Learning Visual Knowledge from Natural Language Supervision

by

Yiwu Zhong

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy
(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN–MADISON

2023

Date of final oral examination: 08/22/2023

The dissertation is approved by the following members of the Final Oral Committee:
Yin Li (Advisor), Assistant Professor, Biostatistics and Medical Informatics
Xiaojin Zhu, Professor, Computer Sciences
Yingyu Liang, Assistant Professor, Computer Sciences
Junjie Hu, Assistant Professor, Biostatistics and Medical Informatics
Lorenzo Torresani, Research Director, Facebook AI Research (FAIR) of Meta

To my family.

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my advisor Prof. Yin Li. Without his continuous guidance and strong support throughout these years, this dissertation would not have come to fruition. I felt quite fortunate to have been under Yin's supervision. His remarkable expertise, deep insights, and commitment to research excellence have been shaping not only the technical aspects of my work, but also my ability to identify meaningful research questions, develop a diverse range of research skills, and plan a path for my future career. Thanks Yin for his invaluable mentorship and patience.

I would like to express my appreciation to the rest of my thesis committee members for their insightful feedback and hard questions. In addition, I would like to thank my mentors across multiple internships, Prof. Liwei Wang, Dr. Dong Yu, Dr. Jianwei Yang, Dr. Jianfeng Gao, Xueting Yan, Dr. Licheng Yu, Dr. Jia Xu, and Prof. Meng Fang, for their valuable guidance and constructive suggestions. I would also like to thank my collaborators for their intriguing discussions and helpful comments: Prof. Chenliang Xu, Dr. Jing Shi, Dr. Pengchuan Zhang, Dr. Chunyuan Li, Dr. Noel Codella, Liunian Li, An Yan and so on.

Further, words cannot adequately express the depth of my gratitude to my beloved family: my parents (Guihong Chen and Jianhong Zhong), my wife's parents (Yonglian Dai and Lianye Li), and my little brother (Yixin Zhong). Specially, I want to thank my beloved wife (Mengtong Li) and our future kids. Their love and belief in me have been the driving force behind my motivation and determination throughout this journey.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Over recent years, deep learning models have achieved remarkable success in visual recognition. However, many of these vision models build on fully-supervised learning and rely on labor-intensive, task-specific annotations that limit their scalability across many categories and tasks. These fully-supervised models thus fall short in open-world visual understanding and are inadequate for numerous real-life applications. To bridge the gaps, my research addresses learning from paired visual and text data, readily available in great abundance over the Internet, in order to build models generalizable to open-world visual concepts across diverse vision tasks. With visual-text pairs as only training data, my work demonstrates, for the first time, that a broad spectrum of visual knowledge can be learned, encompassing object concepts, object relationships, scene components, and action procedures. Further, the learned knowledge can support a variety of vision tasks, ranging from zero-shot object detection, to open-set scene graph generation, to controllable and grounded image captioning, and to zero-shot action classification and forecasting.

1 INTRODUCTION

Driven by large-scale, manually-annotated data, deep learning models achieved unprecedented success in visual recognition. However, many existing vision models build on fully-supervised learning and require labor-intensive, task-specific annotations that are difficult to scale to many categories and tasks. For example, the COCO dataset Lin et al. (2014) — a commonly used object detection dataset, costs more than 70 thousand hours for humans to annotate only 80 object categories, including bounding boxes (*e.g.*, object locations in images) and their categorical labels (*e.g.*, "cat", "bus"). It is difficult to scale up these annotations to thousands of object categories or cover other types of visual concepts (*e.g.*, object relationships, human actions). Meanwhile, fully-supervised learning is limited to the predefined categories in training data, and thus the fully-supervised object detectors trained on the COCO dataset can merely identify those 80 categories. Hence, the fully-supervised methods fall short in addressing open-world visual understanding where the models should be generalizable to broad visual concepts that may not appear during training, thereby being inadequate for real-life applications that require models to understand broad visual concepts, such as household robots and self-driving cars.

The paradigm of fully-supervised learning also significantly differs from human learning ability. Consider the example in Figure 1.1. While "okapi" could be a novel concept to most people, when described as an animal with giraffes' heads and zebras' legs, we can effortlessly recognize this new visual concept of okapi in the images. Humans can learn visual concepts from the text that describe the visual appearance of these concepts. *What if vision models can learn visual concepts from the textual descriptions?*

Millions of visual data and their text descriptions over the Internet present a unique opportunity to address this question. For example, the images paired with Alt-text and the videos paired with audio-transcribed narrations can be readily harvested from the web Sharma et al. (2018); Radford et al. (2021); Jia et al. (2021); Miech et al. (2019). The text descriptions in these visual-text data naturally provide rich associations with various aspects of visual concepts. For instance, the objects in images (*e.g.*, "boy", "kite") are represented by the nouns in text descriptions, and the human actions in videos (*e.g.*, "add butter") are described by the verbs in video narrations. Moreover, free-form text descriptions in large quantities cover diverse visual concepts (*e.g.*, thousands of objects commonly appearing in daily life). Learning

Figure 1.1: Textual descriptions vividly depict the appearance of visual concepts and humans are able to learn new visual concepts from these descriptions.

these concepts from text descriptions can enable the capability of open-vocabulary recognition. Therefore, these abundant visual-text pairs point towards a promising solution to open-world visual understanding, hereby addressing the limitation posed by fully-supervised learning approaches.

Learning from vision-language pairs has long been a topic of interest. While early work primarily harnesses the image-tags pairs Divvala et al. (2014); Joulin et al. (2016), there has been a noticeable attention shift towards learning from visual-text pairs Radford et al. (2021); Miech et al. (2020). With prior works as a reference and foundation, my work addresses the learning of visual knowledge from the visual-text pairs. My work demonstrates, for the first time, that a broad spectrum of visual knowledge can be learned from the consistent source of training data, namely the web-curated visual-text pairs. The learned visual knowledge allows the tackling of a variety of vision problems, such as object detection, scene graph generation, caption generation, and procedural activity understanding. Specifically, my work explores (1) learning to recognize the object concepts in image regions, so as to enable open-vocabulary object detection; (2) learning to discern the relationships among image objects and to generate graphical representations for images; (3) learning to identify the salient scene components so that multiple image captioning capabilities are unified in a single model; (4) learning to reason about action steps and their temporal ordering in videos, going beyond static images.

1.1 Thesis Statement

Vision models, learning from web-curated visual-text pairs, can generalize to diverse visual concepts across a broad spectrum of vision tasks. This paradigm facilitates open-world visual understanding, without the need for exhaustive annotations typically used by fully-supervised methods.

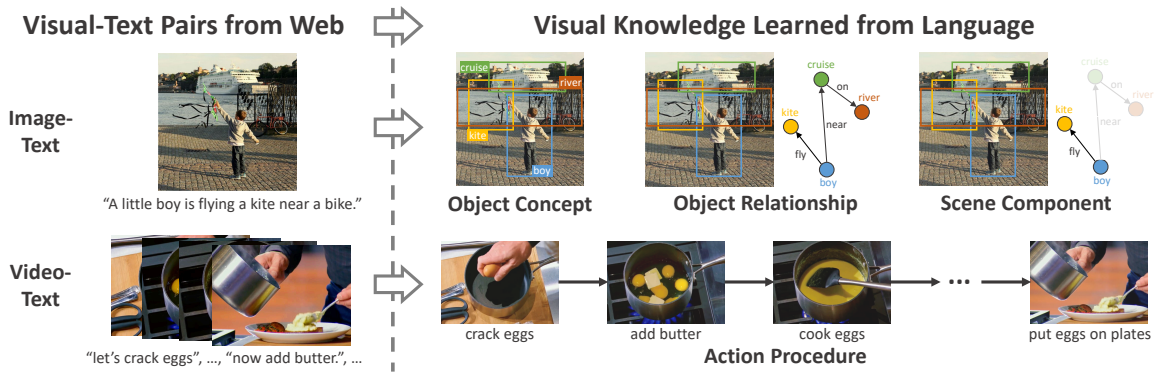


Figure 1.2: Learning visual knowledge from language supervision. The visual-text pairs collected from the web (left) can be used as the only training data to learn a diverse range of visual knowledge (right), including object concepts (*e.g.*, boy) Zhong et al. (2022), object relationships (*e.g.*, fly) Zhong et al. (2021), scene components (*e.g.*, salient sub-graphs) Zhong et al. (2020), and action procedures (*e.g.*, ordered action steps) Zhong et al. (2023).

1.2 Overview

My research goal is to build AI models that learn visual knowledge from language. However, a major obstacle to learning visual knowledge from visual data and their paired text descriptions lies in the absence of fine-grained alignment between vision and language. For example, image captions often provide a coarse description of the image content, making it unclear which specific image regions correspond to the entities mentioned in the caption. Likewise, videos and their narrations often exhibit loose temporal alignment. Narrators typically discuss future steps before executing them, and sometimes the narrated steps are omitted in the corresponding videos. Such imprecise alignment poses challenges in learning action steps and their temporal ordering. The lack of fine-grained alignment between visual entities and text spans makes it challenging to acquire visual knowledge that is intrinsically linked to the corresponding visual appearance.

To address this challenge, I have worked on methods to establish the fine-grained alignment between vision and language, for learning object concepts Zhong et al. (2022), object relationships Zhong et al. (2021), scene components Zhong et al. (2020), and action procedures Zhong et al. (2023). Figure 1.2 provides an illustrative example. The accompanying text description allows the models to learn a broad spectrum of visual knowledge, such as object concepts (*e.g.*, "boy", "kite"), their relationships in images (*e.g.*, "flying" between "boy" and "kite"), scene components (*e.g.*, salient

sub-graphs), and the ordered actions to perform a task (*e.g.*, steps to make scrambled eggs). Overall, my work demonstrates that a vast range of visual knowledge can be learned from human language using the same source of training data (*i.e.*, visual-text pairs from the web), without the requirement of label-intensive, task-specific annotation. Moreover, for each aspect of visual knowledge, the models are capable of identifying all concepts that ever appeared in the text, thereby achieving open-world visual understanding. More broadly, my research also addresses the problems of grounding image regions to human-annotated scene graphs Shi et al. (2021a) and to phrases in sentences Li et al. (2022), and the problem of building recognition models that can be interpreted by language-described visual attributes Yan et al. (2023).

My prior work relevant to this dissertation can be organized into four main aspects: learning object concept (Chapter 3), learning object relationship (Chapter 4), learning scene component (Chapter 5), and learning action procedure (Chapter 6). I present a brief introduction of each chapter in this section.

Learning object concept

Chapter 3 addresses open-vocabulary object detection — a fundamental vision problem. I propose a novel method RegionCLIP to learn region-level visual representations from image-text pairs. The learned representations enable object detection with an open vocabulary of object concepts, by matching the representation of each region to the representation of text templates filled by any object concept. RegionCLIP demonstrates new state-of-the-art results on open-vocabulary object detection and presents strong results on zero-shot inference for object detection.

Learning object relationship

Moving beyond individual object concepts, Chapter 4 considers the detection of the relationships between objects, organized into a graphical abstraction — known as image scene graph Krishna et al. (2017); Xu et al. (2017); Zellers et al. (2018), consisting of localized and categorized objects (nodes) and the relationships between objects (edges). Image scene graph serves as an important representation for many vision tasks including action recognition Ji et al. (2020), 3D scene understanding Armeni et al. (2019); Wald et al. (2020), image generation and editing Johnson et al. (2018); Dhamo et al. (2020), and vision-language tasks (*e.g.*image captioning Yang et al.

(2019); Yao et al. (2018); Zhong et al. (2020) and visual question answering Shi et al. (2019); Teney et al. (2017); Hudson and Manning (2019))

In Chapter 4, I present the first method to learn object relationships from image-text pairs without using manual annotations, so that the trained model can take an image as input and output a scene graph. Our method trained by only image-text pairs can even outperform the latest method trained by human-annotated scene graphs. These results suggest that our method can generate scene graph with high quality by learning from image-text pairs.

Learning scene component

Built on image scene graph, Chapter 4 addresses the problem of discovering and describing meaningful scene components of an image, so as to generate diverse text descriptions that are grounded into and controlled by local image regions, unifying previous works on diverse Wang et al. (2017); Deshpande et al. (2019), grounded Selvaraju et al. (2019); Zhou et al. (2019) and controllable image captioning Lu et al. (2018); Cornia et al. (2019).

In Chapter 5, I propose a novel image captioning method that takes an image as input, and generates diverse and controllable captions, with the nouns in captions grounded to the image regions. The key idea is to leverage the scene graph generated for each image, to identify the meaningful sub-graphs (scene components) within scene graph, and to decode each sub-graph into a sentence description. The design of selecting scene components achieves state-of-the-art results on several image captioning benchmarks simultaneously, including diversity, grounding and controllability.

Learning action procedure

The visual world in real life is essentially dynamic and can be recorded in videos. For example, instructional videos from the Internet (*e.g.*, YouTube) capture individual actions and the procedures to perform tasks (*e.g.*, cooking, device installation) Miech et al. (2019); Tang et al. (2019b).

In Chapter 6, I propose a video-and-language pretraining framework for learning procedure-aware video representation from millions of instructional videos and their narrations, without the need for human annotations. Our method jointly learns a video encoder that captures concepts of action steps, as well as a diffusion model

that reasons about the temporal dependencies among steps. The resulting video representations establish new state-of-the-art results on both step classification and forecasting tasks across the major benchmarks. Importantly, our model supports zero-shot inference and is capable of generating diverse step predictions.

1.3 Contributions

The contributions of this dissertation are summarized as follows:

- To learn object concepts, I proposed a novel method RegionCLIP which learns region-level visual representations from text descriptions. Without the need of exhaustive human annotation for object concepts, the learned representations support open-vocabulary object detection with many object concepts and demonstrate new state-of-the-art results.
- I showed that for the first time, object relationships, organized as image scene graphs, can be learned from image-text pairs. Our method does not require human-annotated scene graphs for training, while achieving superior results than those trained by annotated scene graphs.
- I introduced the first image captioning method that unifies multiple capabilities in a single model, including accurate, diverse, grounded and controllable captioning. The key is to learn scene components from image-text pairs by selecting the sub-graphs from the input scene graph.
- I present a novel video-language pretraining method that learns video representations from millions of instructional videos and their narrations. For the first time, our method captures the action ordering and its variations from unannotated videos, without the help of human annotation. Beyond new state-of-the-art results on major benchmarks, the resulting video representations additionally enable new capabilities, including zero-shot step forecasting and diverse step predictions.

2 BACKGROUND

In this chapter, I review the related work of my dissertation, including the parts below:

- **Learning Visual Knowledge from Language**: This section provides an overview of previous work of leveraging language descriptions for visual learning, covering multiple domains. They are further elaborated in the following sections.
- **Visual Representation Learning**: Related to Chapter 3 of learning object concepts, this section provides a survey of visual representation learning for images and image regions.
- **Zero-shot and Open-vocabulary Object Detection**: Related to Chapter 3 of learning object concepts, this section introduces the latest work of object detection. These methods seek to detect novel object categories that have not been seen during training.
- **Scene Graph Generation**: Related to Chapter 4 of learning object relationships, this section shows the previous methods that were built for learning visual relationships, with various levels of annotation requirements, including: localized scene graphs and unlocalized scene graphs.
- **Image Captioning**: Related to Chapter 5 of learning scene components, this section reviews the previous work that was designed for each individual task within image captioning, including: caption quality, caption grounding, caption diversity and controllability.
- **Video Representation Learning**: Related to Chapter 6 of learning action procedures, this section summarizes the related work that learns video representations from instructional videos for understanding procedural activities.
- **Diffusion Models**: This section presents previous work on the diffusion models. It is used to model the procedure variations in Chapter 6.

2.1 Learning Visual Knowledge from Language

The availability of image-text pairs and video-text pairs on the Internet has spurred a surge of interest in learning from these visual-text pairs. Early work focused on

learning from image-hashtag pairs for visual representation learning Chen and Gupta (2015); Joulin et al. (2016) and for recognizing objects, scenes, and actions Chen et al. (2013); Divvala et al. (2014); Farhadi et al. (2009); Lampert et al. (2009). More recent work has shifted attention to learning from images/videos and their sentence descriptions. For example, visual-sentence pairs were used (1) for visual representation learning, via image-text matching Radford et al. (2021), video-text matching Miech et al. (2020), image captioning Desai and Johnson (2021), or image-conditioned language modeling Sariyildiz et al. (2020), and (2) for visual-textual joint representation learning, using context prediction tasks Chen et al. (2020b); Li et al. (2020b); Tan and Bansal (2019); Lu et al. (2019); Sun et al. (2019); Zhu and Yang (2020); Zellers et al. (2021). Image descriptions were also exploited for object recognition Wang et al. (2009), object detection Ye et al. (2019); Jerbi et al. (2020); Yang et al. (2018b); Zareian et al. (2021), and the task of image captioning Hossain et al. (2019); Karpathy and Fei-Fei (2015); Vinyals et al. (2015); Donahue et al. (2015); Xu et al. (2015).

My dissertation lies in the same area of learning visual knowledge from image/video-text pairs and seeks to learn object concepts, object relationships, scene components, and action procedures. Inspired by the recent success of learning image-level representation via image-text matching pretraining Radford et al. (2021), my work in Chapter 3 extends this matching idea to learn *region-level* visual representation and thus enables *open-vocabulary* object detection for many object concepts. Beyond individual object concepts, my work in Chapter 4 *for the first time*, learns to detect scene graphs for the input images, with image-text pairs as only training data. Further, by learning scene components from image-text pairs, my work in Chapter 5 is the *first* image captioning method that enables multiple essential abilities *at the same time*, including diverse, controllable and grounded captioning. Going beyond static images, my work in Chapter 6 focuses on learning procedure-aware video representations from instructional videos and their narrations and enables zero-shot inference for both step classification and step forecasting.

2.2 Visual Representation Learning

Representation learning for images

Early work on visual representation learning focused on training image classification models using labor-intensive human annotations Krizhevsky et al. (2012); Simonyan

and Zisserman (2015); Szegedy et al. (2015); He et al. (2016); Dosovitskiy et al. (2021). The learned features can be transferred to recognition tasks Girshick et al. (2014), and the classifier can be used to label images for semi-supervised learning Pham et al. (2021); Xie et al. (2020); Yalniz et al. (2019). To reduce the annotation burden, self-supervised learning He et al. (2020); Chen et al. (2020a); Grill et al. (2020); Caron et al. (2020) has received considerable attention recently. The representation learned by self-supervised methods is limited to the vision modality and can not be directly interpreted by natural language.

The most relevant work in this direction is learning visual representations from natural language, such as image tags Hironobu et al. (1999); Barnard et al. (2003); Divvala et al. (2014); Chen and Gupta (2015); Joulin et al. (2016) and text descriptions Wang et al. (2009); He and Peng (2017); Sariyildiz et al. (2020); Desai and Johnson (2021); Zhong et al. (2021). Leveraging millions of image-text pairs collected from the Internet, recent methods in vision-language pretraining Radford et al. (2021); Jia et al. (2021) learned to match images with text descriptions and demonstrated impressive performance on zero-shot inference and transfer learning for image classification. However, these methods focus on global representation tailored for image classification. In Chapter 3, I propose to learn visual representation for local image regions so as to enable zero-shot inference and transfer learning for region-based reasoning (*e.g.*, object detection).

Representation learning for image regions

Many region-based reasoning tasks, such as object detection Ren et al. (2015a); Redmon et al. (2016); Tian et al. (2019); Carion et al. (2020), rely on dense human annotations Everingham et al. (2010); Lin et al. (2014); Krishna et al. (2017); Gupta et al. (2019). Recently, semi-supervised learning was explored Xu et al. (2021b); Zoph et al. (2020); Sohn et al. (2020), where pretrained detectors are used to create pseudo labels of image regions. Beyond object labels, region representation learning benefits from additional labels of object attributes Krishna et al. (2017); Anderson et al. (2018); Zhang et al. (2021), showing noticeable improvement on vision-language tasks Yu et al. (2021); Chen et al. (2020b); Li et al. (2020b); Tan and Bansal (2019); Zhou et al. (2020); Lu et al. (2019). However, these methods heavily rely on manual annotations and are limited to predefined categories. As a partial remedy, self-supervised learning was extended to region representations Ramanathan et al. (2021); Hénaff et al. (2021).

In Chapter 3, inspired by CLIP Radford et al. (2021) yet distinct from prior work, I propose to learn region representation via vision-language pretraining. The learned representations enable the recognition of many visual concepts within image regions.

2.3 Zero-shot and Open-vocabulary Object Detection

Zero-shot Object Detection

Zero-shot object detection aims at detecting novel object classes that are not seen during detector training Bansal et al. (2018); Rahman et al. (2020b); Zareian et al. (2021); Gu et al. (2021); Zhu et al. (2020); Rahman et al. (2020a). Bansal *et al*. Bansal et al. (2018) learned to match the visual features of cropped image regions to word embeddings Pennington et al. (2014) using max-margin loss. Rahman *et al*. Rahman et al. (2020a) proposed polarity loss to model background category and to cluster categories with similar semantics. Zhu *et al*. Zhu et al. (2020) explored improving localization performance for novel categories by synthesizing visual features with a generative model. These zero-shot object detectors usually rely on the semantic space of pretrained word embeddings Pennington et al. (2014).

Open-vocabulary Object Detection

Recently, Zareian *et al*. Zareian et al. (2021) proposed OVR for open-vocabulary object detection, where a visual encoder is first pretrained on image-text pairs to learn object concepts and then transferred to zero-shot object detection setting. Another close work is ViLD Gu et al. (2021) that focuses on learning object detectors by distilling visual features from a pretrained CLIP model Radford et al. (2021), yet still requires object labels and boxes for training.

In Chapter 3, I present my work that learns object concepts from textual descriptions. Similar to OVR and ViLD, our object detector also leverages the visual-semantic space learned from vision-language pretraining. Different from OVR, I propose to learn region region representations from our "pseudo" region-text pairs given by a pretrained CLIP model. Our method is thus not restricted to existing text descriptions of an image. Unlike ViLD, my work addresses the problem of region representation learning, and focuses on *pretraining* from region-text pairs. As a result, the learned representations support zero-shot inference, while ViLD can not.

2.4 Scene Graph Generation

Fully Supervised Scene Graph Generation

An image scene graph represents localized object instances as nodes and their relationships as edges on the graph. Scene graph generation (SGG) aims to extract this graphical representation from an input image. A related problem is visual relationship detection (VRD) Yu et al. (2017); Lu et al. (2016); Zhang et al. (2017a); Dai et al. (2017b) that also localizes objects and recognizes their relationships yet without the notation of a graph.

Thanks to the development of large-scale densely annotated datasets of image scene graphs, such as Visual Genome (VG) dataset Krishna et al. (2017), a large array of methods have been proposed for scene graph generation. Several different models have been explored, including iterative message passing Xu et al. (2017); Li et al. (2017), recurrent network Zellers et al. (2018), tree structure encoding Tang et al. (2019a); Wang et al. (2020), graph convolution and pruning Li et al. (2018b); Yang et al. (2018a), casual reasoning Chen et al. (2019a); Tang et al. (2020a) and contrastive learning Zhang et al. (2019). A major drawback of these approaches is the requirement of human-annotated, localized scene graphs with categorical labels and locations of all nodes and edges. In Chapter 4, I introduce my work that seeks to address this drawback by learning scene graphs from only image-sentence pairs.

Weakly-supervised Scene Graph Generation

Several recent works have explored weakly supervised settings for VRD Peyre et al. (2017); Baldassarre et al. (2020); Zhang et al. (2017b) and SGG Zhang et al. (2017b); Zareian et al. (2020); Shi et al. (2021b). Most of them addressed the task of VRD and seeks to learn from unlocalized subject-predicate-object (SPO) triplets. For example, Peyre *et al*.Peyre et al. (2017) proposed to assign image-level labels to pairs of detected objects via discriminative clustering. Baldassarre *et al*.Baldassarre et al. (2020) first predicted visual predicates given the detected objects, and then retrieved the subjects and objects using backward explanation techniques. Zhang *et al*.Zhang et al. (2017b) designed a fully convolutional network to jointly learn object detection and predicate prediction from image-level labels, using object proposals as model inputs. They reported results on both VRD and SGG.

The most relevant work is from Zareian *et al*.Zareian et al. (2020). They proposed to learn from unlocalized scene graphs for SGG, and developed a message passing mechanism to update features of detected objects and to gradually refine labels of objects and predicates. A recent work Shi et al. (2021b) presented a simple baseline for weakly supervised SGG using first-order graph matching. Similar to these approaches, my work presented in Chapter 4 explores learning using less labels for SGG. Unlike previous approaches, my method leverages image captions — a different type of labels that are easier to obtain than unlocalized SPO triplets or scene graphs. A concurrent work from Ye *et al*.Ye and Kovashka (2021) also explored learning scene graph from image-sentence pairs. They proposed to use visual grounding to iteratively match the detected image regions and the text entities parsed from captions. Unlike their method, my work in Chapter 4 leverages an object detector to create the pseudo labels for SPO triplets, leading to significantly better empirical results. My work is thus among the first methods to learn scene graphs from only image-sentence pairs.

2.5 Image Captioning

Conventional Image Captioning

Major progress has been made in image captioning Hossain et al. (2019). An encoder-decoder model is often considered, where Convolutional Neural Networks (CNNs) are used to extract global image features, and Recurrent Neural Networks (RNNs) are used to decode the features into sentences Karpathy and Fei-Fei (2015); Vinyals et al. (2015); Donahue et al. (2015); Xu et al. (2015); You et al. (2016); Lu et al. (2017); Rennie et al. (2017); Liu et al. (2018). Object information has recently been shown important for captioning Yin and Ordonez (2017); Wang et al. (2018). Object features from an object detector can be combined with encoder-decoder models to generate high quality captions Anderson et al. (2018).

Several recent works have explored objects and their relationships, encoded in the form of scene graphs, for image captioning Yao et al. (2018); Yang et al. (2019). The most relevant work is Yao et al. (2018). Their GCN-LSTM model used a graph convolutional network (GCN) Kipf and Welling (2016) to integrate semantic information in a scene graph. And a sentence is further decoded using features aggregated over the full scene graph. Similar to Yao et al. (2018), my work in Chapter 5 also uses a GCN for an input scene graph. However, my work learns to select sub-graphs within

the scene graph, and to decode sentences from top-ranked sub-graphs instead of the full scene graph. This design is capable of producing diverse and controllable sentences that are previously infeasible Yao et al. (2018); Yang et al. (2019).

Grounded Captioning

A major challenge of image captioning is that recent deep models might not focus on the same image regions as a human would when generating each word, leading to undesirable behaviors, e.g., object hallucination Rohrbach et al. (2018); Das et al. (2017). Several recent work Xu et al. (2015); Anderson et al. (2018); Selvaraju et al. (2019); Zhou et al. (2019); Ma et al. (2019); Johnson et al. (2016); Yang et al. (2017) has been developed to address the problem of grounded captioning—the generation of captions and the alignment between the generated words and image regions. My work presented in Chapter 5 follows the weakly supervised setting for grounded captioning, where we assume that only the image-sentence pairs are known. The key innovation is to select a sub-graph from an image scene graph for sentence generation, thus constraining the grounding within the sub-graph.

There is other relevant work on generating text descriptions of local image regions, also known as dense captioning Johnson et al. (2016); Yang et al. (2017); Kim et al. (2019). Both my work in Chapter 5 and dense captioning methods can create localized captions. The key difference is that my method aims to generate sentence descriptions of scene components that spans multiple image regions, while dense captioning methods focused on generating phrase descriptions for individual regions Johnson et al. (2016); Yang et al. (2017) or pairs of local regions Kim et al. (2019).

Diverse and Controllable Captioning

The generation of diverse and controllable image descriptions has also received considerable attention. Several approaches have been proposed for diverse captioning Shetty et al. (2017); Li et al. (2018a); Dai et al. (2017a); Vijayakumar et al. (2018); Wang et al. (2017); Deshpande et al. (2019); Aneja et al. (2019). Wang *et al*. Wang et al. (2017) proposed a variational auto-encoder that can decode multiple diverse sentences from samples drawn from a latent space of image features. This idea was further extended by Aneja et al. (2019), where every word has its own latent space. Moreover, Deshpande *et al*. Deshpande et al. (2019) proposed to generate various

sentences controlled by part-of-speech tags. There is a few recent work on controllable captioning. Lu *et al*. Lu et al. (2018) proposed to fill a generated sentence template with the concepts from an object detector. Cornia *et al*. Cornia et al. (2019) used grounding annotations to select object regions and these regions were further used to predict corresponding textual chunks for diverse and controllable captioning. Similar to Cornia et al. (2019), my work introduced in Chapter 5 addresses diversity and controllability within the same model. Different from Cornia et al. (2019), my work is trained using only image-sentence pairs and can provide additional capacity of caption grounding.

Graph Partitioning

My work in Chapter 5 explores the decomposition of scene graphs into sub-graphs for generating accurate, diverse, and controllable captions. Similar graph partitioning problem has been previously considered in vision for image segmentation Jianbo Shi and Malik (2000); Felzenszwalb and Huttenlocher (2004) and visual tracking Tang et al. (2015); Song et al. (2019), but has not been explored for image captioning.

2.6 Video Representation Learning

Learning Video Representation from Instructional Videos

The success of vision-and-language pre-training has fueled a new line of research that seeks to learn concepts of individual steps from instructional videos and their narrations Malmaud et al. (2015); Xu et al. (2020); Shen et al. (2021); Han et al. (2022). For example, Miech *et al*. Miech et al. (2020) propose MIL-NCE to learn video representations from instructional videos Miech et al. (2019) and their narrations extracted using ASR.

The most relevant work is DistantSup Lin et al. (2022), where they propose using distant supervision from a textual knowledge base (wikiHow) Koupaee and Wang (2018) to denoise text narrations from ASR. Specifically, DistantSup leverages a pre-trained language model Song et al. (2020) to link step descriptions from wikiHow to text narrations from video ASR results, and thus to create training labels for individual steps in videos. Different from Lin et al. (2022), the method I proposed in Chapter 6 models the temporal ordering of steps in procedural activities, and thus moves beyond

representations of single steps to support temporal reasoning in videos. Further, our method learns from videos and narrations only, with the help of a pre-trained image-language model Radford et al. (2021) yet without using a textual knowledge base like wikiHow Koupaee and Wang (2018).

Video-and-Language Pre-Training

A relevant topic is video-and-language pre-training, aiming at learning video representation from videos and their paired natural language descriptions Ghadiyaram et al. (2019); Xu et al. (2021a); Bain et al. (2021); Lei et al. (2021); Yang et al. (2021); Sun et al. (2019); Zhu and Yang (2020); Li et al. (2020a); Luo et al. (2020); Zellers et al. (2021); Fu et al. (2021); Wang et al. (2022), often generated from ASR outputs. Despite the latest development in ASR, automatically-transcribed speech from videos can be rather noisy and lacks precise temporal alignment with the visual content. Several recent works seek to address this challenge. VideoCLIP Xu et al. (2021a) starts from the pre-trained MIL-NCE model and further improves the model by retrieval augmented training with overlapped video-text pairs. Bain *et al*. Bain et al. (2021) collect a less noisy dataset of video alt-text pairs and geared the model to match these pairs. My work in Chapter 6 shares the key idea of learning from video and text data as prior work, and seeks to leverage external knowledge from a pre-trained image-language model Radford et al. (2021).

Another relevant work is MERLOT Zellers et al. (2021). While both MERLOT and my work seek to learn video representation, our method differs from MERLOT in two folds. Our method models the sequence order of video clips for understanding procedural activities. MERLOT learns binary relative order between two given video frames for multi-modal reasoning and does not directly support action forecasting. Both methods consider a masked prediction task, yet MERLOT predicts the most likely text embeddings, while our method estimates the distribution of video representations using a deep probabilistic model.

Understanding Procedural Activities

Reasoning about procedural activities, including their action steps and the temporal ordering of these steps, has been a central problem in activity recognition. While early works model temporal ordering with stochastic grammars Pei et al. (2011);

Gupta et al. (2009); Ryoo and Aggarwal (2006); Ivanov and Bobick (2000); Brand et al. (1997); Nevatia et al. (2003), more recent works consider supervised learning to localize steps and predict their ordering by learning from videos with human annotated action steps Kuehne et al. (2014); Zhou et al. (2018b); Elhamifar and Naing (2019); Zhukov et al. (2019); Tang et al. (2019b); Chang et al. (2020); Damen et al. (2021). To alleviate the burden of costly video annotations, several works propose various forms of weakly supervised settings, with assumptions that the ordered list of steps is given without their temporal boundaries Bojanowski et al. (2014, 2015); Zhukov et al. (2019); Zhao et al. (2022), or that the key steps and their ordering remain fixed across all videos Sener et al. (2015); Alayrac et al. (2016); Goel and Brunskill (2019); Kukleva et al. (2019); Elhamifar and Huynh (2020).

Most of prior methods focus on the tasks of step classification and localization Bojanowski et al. (2014, 2015); Zhukov et al. (2019); Sener et al. (2015); Alayrac et al. (2016); Kukleva et al. (2019); Elhamifar and Naing (2019); Elhamifar and Huynh (2020). Others have considered the tasks of step forecasting Sener and Yao (2019), step verification Qian et al. (2022) and procedure planning Zhao et al. (2022). In Chapter 6, I introduce my work that also seeks to understand procedural activities. Different from these approaches, our method focuses on learning video representation from videos and their narrations without using human annotations. The resulting video representation can be leveraged for step classification and step forecasting.

2.7 Diffusion Models

Diffusion models Sohl-Dickstein et al. (2015); Song and Ermon (2020) provide a powerful approach to characterize the probability density of high dimensional signals, and have recently demonstrated impressive results on generating high fidelity visual data, such as images Nichol et al. (2022); Ramesh et al. (2022); Saharia et al. (2022); Rombach et al. (2022), videos Ho et al. (2022), and human body motion Tevet et al. (2023). My work in Chapter 6 adapts the diffusion process to model the temporal ordering of steps in procedural videos. In doing so, our method not only facilities the learning of expressive video representations for individual steps, but also enables the anticipation of future action steps.

3 LEARNING OBJECT CONCEPT

The recent advances in vision-language representation learning has created remarkable models like CLIP Radford et al. (2021), ALIGN Jia et al. (2021) and Florence Yuan et al. (2021). Such models are trained using hundreds of millions of image-text pairs by matching images to their captions, achieving impressive results of recognizing a large set of concepts without manual labels, and capable of transferring to many visual recognition tasks. Following their success on image classification, a natural question is whether these models can be used to reason about image regions, *e.g.*, for tasks like object detection.

To answer this question, we construct a simple R-CNN style Girshick et al. (2014) object detector using a pretrained CLIP model, similar to adapting a convolutional network pretrained on ImageNet. This detector crops candidate object regions from an input image, and applies the CLIP model for detection by matching visual features of cropped regions to text embeddings of object categories. Fig. 3.1(a-b) shows the results on LVIS dataset Gupta et al. (2019). When using object proposals Ren et al. (2015a) as the input regions, scores from CLIP often fail to capture the localization quality (Fig. 3.1a). Even with ground-truth object boxes, classification accuracy using CLIP drops significantly from 60% on ImageNet to 19% on LVIS, with a similar number of classes (Fig. 3.1b). There is thus a major performance degradation when applying a pretrained CLIP model for object detection. *How can we empower a vision-language pretrained model to reason about image regions?*

We believe the main gap lies in the training of these vision-language models. Many existing vision-language models, including CLIP, are trained to match an image with its image-level text description. The training is unaware of the alignment between local image regions and text tokens. Thus, the models are unable to precisely ground a textual concept to an image region. Further, cropping local image regions and matching them to text tokens largely ignore the surrounding visual context that is critical for object recognition, not to mention the high computational cost, *e.g.*a few seconds per image on a modern GPU.

In this chapter, we explore learning *region representations* via vision-language pretraining and the learned representations support object detection with many object concepts. Our key idea is to explicitly align image regions and text tokens during pretraining. However, two key challenges arise. First, the fine-grained alignment

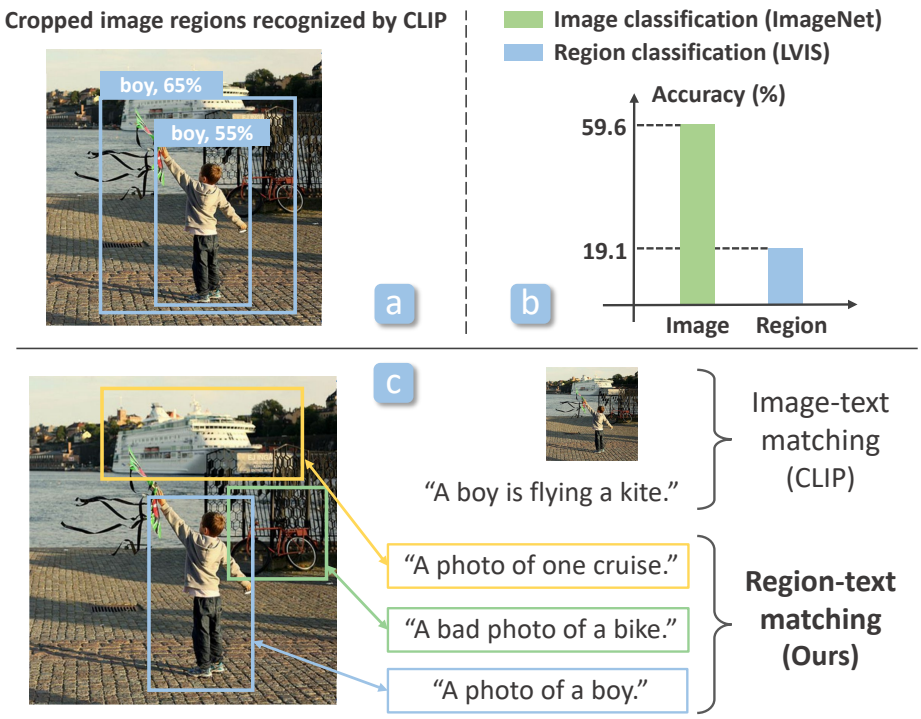


Figure 3.1: (**a**). A pretrained CLIP model Radford et al. (2021) failed to capture localization quality. (**b**). A major drop on accuracy when using the same pretrained CLIP to classify image regions. (**c**). Our key idea is learning to match *image regions* and their text descriptions.

between image regions and text tokens is not available in image-text pairs and expensive to annotate. Second, the text description of an image is often incomplete, *i.e.*many image regions are not described by the text.

To address these challenges, we propose to bootstrap from a pretrained vision-language model (*e.g.*, CLIP) to align image regions and text tokens, and to fill in the missing region descriptions, as illustrated in Fig. 3.1c. Specifically, our method starts with a pool of object concepts parsed from text corpus, and synthesizes region descriptions by filling these concepts into pre-defined templates. Given an input image and its candidate regions from either object proposals or dense sliding windows, a pretrained CLIP model is used to align the region descriptions and the image regions, creating "pseudo" labels for region-text alignment. Further, we combine "pseudo" region-text pairs and ground-truth image-text pairs to pretrain our vision-language model via contrastive learning and knowledge distillation. Although the "pseudo" region-text pairs are noisy, they still provide useful information for learning region

representations, and thus help to bridge the gap in object detection, as validated by our experiments.

This chapter is organized as follows. In Section 3.1, I summarize our contributions. In Section 3.2 and Section 3.3, I present our method and key experiments, respectively. Finally, I conclude the chapter in Section 3.4.

3.1 Contributions

The contributions of this chapter are summarized into three folds:

- We propose a novel vision-language pretraining method for learning visual region representations. The learned representations support image region recognition with many object concepts.
- A key technical innovation that facilitates our pretraining is a scalable approach using text prompts to align the object descriptions with image regions, without relying on human annotations nor limited to the text paired with an image.
- Our pretrained model presents new state-of-the-art results results when transferred to open-vocabulary object detection, and demonstrates promising capability on zero-shot inference for object detection.

This work was a collaboration with Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, and Jianfeng Gao. The work has been accepted by CVPR 2022 with title as "RegionCLIP: Region-based Language-Image Pretraining" Zhong et al. (2022).

3.2 Method

Our goal is to learn a regional visual-semantic space that covers rich object concepts so that it can be used for open-vocabulary object detection. Consider a text description t that describes the content of region r in an image I. In the visual-semantic space, the visual region representation $V(I, r)$ extracted from r should be matched to text representation $L(t)$. V is a visual encoder that takes image I and a region location r, and outputs a visual representation for this region. L is a language encoder that converts a text description in natural language to a semantic representation.

Disentanglement of recognition and localization. There are two key components for region-based reasoning: localization and recognition. Inspired by Singh et al. (2018), we disentangle these two components, use existing region localizers, and consider a recognition problem. Our focus is thus learning visual-semantic space to recognize image regions without human annotations.

Model overview. As shown in Fig. 3.2, we denote V_t and L as visual and language encoders pretrained to match images to their descriptions, such as CLIP. Our goal is to train a visual encoder V so that it can encode image regions and match them to region descriptions encoded by language encoder L. To address the challenge of missing region descriptions, as shown at the bottom of Fig. 3.2, we construct a pool of object concepts, create the region descriptions by filling concepts into prompts, and leverage a teacher encoder V_t to align these text descriptions with the image regions proposed by an image region localizer. Given the created region-text pairs, our visual encoder V learns to match these pairs via contrastive learning and concept distillation. Once pretrained, our model supports zero-shot inference for region recognition, and can be transferred to train object detector when the human annotation is available. We now describe region-level visual and semantic representations, and the alignment between image regions and text descriptions.

Visual and Semantic Region Representation

Visual region representation. Image regions can be proposed by either off-the-shelf object localizers (*e.g.*, RPN Ren et al. (2015a)) or dense sliding windows . By default, we use RPN pretrained on human-annotated object bounding boxes *without* object labels. We use RPN to propose image regions and obtain N image regions, denoted as $\{r_i\}_{i=1,...,N}$.

Given the proposed regions, the visual representation $\mathbf{v}_i$ of region r_i is extracted from our visual encoder V with a feature pooling method, such as RoIAlign He et al. (2017). RoIAlign pools regional visual features from the feature map of a full image by using interpolation. We note that our visual encoder V is initialized by the teacher V_t so that it can have a good starting point in visual-semantic space.

Semantic region representation. A single image usually contains rich semantics, covering one or more objects from thousands of categories. It is costly to annotate all these categories in the large-scale image-text datasets. To this end, we first build a large pool of concepts to exhaustively cover regional concepts. As shown at the

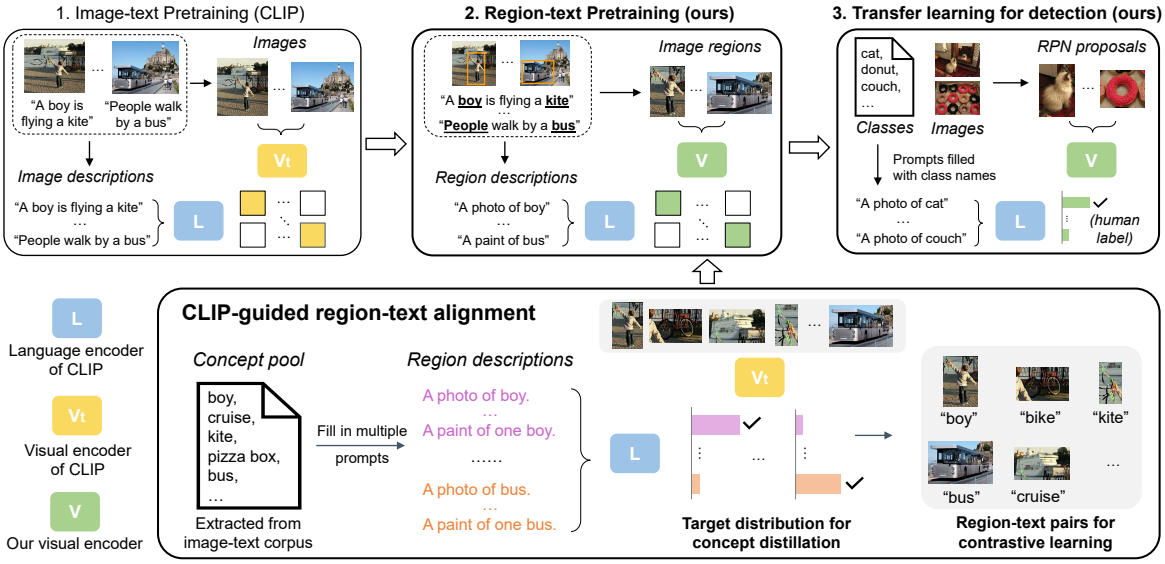


Figure 3.2: Method overview. We propose to learn visual representations for image regions via vision-language pretraining. Panel 1: With contrastive learning, CLIP is able to match images and their descriptions. Panel 2: Initialized by pretrained CLIP, our visual encoder learns visual region representations from the created region-text pairs. Specifically, as shown in the bottom row, we first create texts by filling the prompts with object concepts which are parsed from image descriptions, then use pretrained CLIP to align these texts and image regions proposed by RPN. Panel 3: When human annotation for image regions is available, we transfer our visual encoder for object detection.

bottom of Fig. 3.2, we create a pool of object concepts which are parsed from text corpus (*e.g.*, the image descriptions collected from the Internet), by using off-the-shelf language parsers Jiayuan and Seito (2018); Schuster et al. (2015).

Given the concept pool, the semantic representations for regions are created by two steps: (1) a short sentence for each concept is created by filling it to prompt templates (*e.g.*, prompts of CLIP Radford et al. (2021)), *e.g.*, the "kite" concept is converted to "A photo of a *kite*"; (2) the resulting text descriptions are further encoded into semantic representations by using the pretrained language encoder L. Finally, all regional concepts are represented by their semantic embeddings $\{\mathbf{l}_j\}_{j=1,...,C}$ and C denotes the size of concept pool.

While our region descriptions are built on existing image descriptions, our method is not constrained by the particular text descriptions that pair with images. Importantly, using a powerful language encoder L trained with hundreds of millions of text

descriptions containing tens of thousands of words allows us to easily customize and scale up our concept pool. Such a capacity is deemed difficult to achieve using human annotations. In addition, the disentanglement of visual recognition and localization makes our method flexible to adopt different ways of extracting candidate regions.

Visual-Semantic Alignment for Regions

Alignment of region-text pairs. We leverage a teacher visual encoder V_t to connect image regions and our created texts (represented as semantic embeddings). Again, visual representation $\mathbf{v}_i^t$ of region r_i is extracted from teacher encoder V_t by pooling features from a local image region with RoIAlign. A matching score $S(\mathbf{v}, \mathbf{l})$ between $\mathbf{v}_i^t$ and each concept embedding $\mathbf{l}_j$ is then computed by

$$S(\mathbf{v}, \mathbf{l}) = \frac{\mathbf{v}^T \cdot \mathbf{l}}{||\mathbf{v}|| \cdot ||\mathbf{l}||}. \tag{3.1}$$

The object concept with highest matching score, denoted as $\mathbf{l}_m$, is selected and linked to region r_i. Finally, we obtain a pseudo label for each region, forming the pairs of $\{\mathbf{v}_i, \mathbf{l}_m\}$.

Our pretraining scheme. Our pretraining leverages both created region-text pairs and the existing image-text pairs. Given the aligned region-text pairs ($\{\mathbf{v}_i, \mathbf{l}_m\}$), we design a contrastive and a distillation loss based on the regions across different images to pretrain our visual encoder. Inspired by Oord et al. (2018), the contrastive loss is computed as

$$L_{cntrst} = \frac{1}{N} \sum_i -\log(p(\mathbf{v}_i, \mathbf{l}_m)), \tag{3.2}$$

where $p(\mathbf{v}_i, \mathbf{l}_m)$ is given by

$$p(\mathbf{v}_i, \mathbf{l}_m) = \frac{\exp(S(\mathbf{v}_i, \mathbf{l}_m)/\tau)}{\exp(S(\mathbf{v}_i, \mathbf{l}_m)/\tau) + \sum_{k \in \mathcal{N}_{r_i}} \exp(S(\mathbf{v}_i, \mathbf{l}_k)/\tau).} \tag{3.3}$$

Here τ is a predefined temperature, and $\mathcal{N}_{r_i}$ represents a set of negative textual samples for region r_i, *i.e.*, the object concepts that are not matched to region r_i but matched to other regions in the batch.

Since positive pairs in the contrastive loss are inevitably "noisy", we also consider knowledge distillation for image regions. Knowledge distillation learns from a soft target and helps to handle the noise in those pseudo region-text pairs. This distillation

loss is defined as

$$L_{dist} = \frac{1}{N} \sum_{i} L_{KL}(\mathbf{q}_i^t, \mathbf{q}_i), \tag{3.4}$$

where L_{KL} is the KL divergence loss; both $\mathbf{q}_i^t$ and $\mathbf{q}_i$ are probabilities over all object concepts. $\mathbf{q}_i^t$ is a soft target from teacher model computed as $softmax(S(\mathbf{v}_i^t, \mathbf{l}_1)/\tau, ..., S(\mathbf{v}_i^t, \mathbf{l}_C)/\tau)$. $\mathbf{q}_i$ is similarly computed from our student model.

Given image-text pairs collected from the Internet, our region-level contrastive loss L_{cntrst} can naturally extend to image-level contrastive loss $L_{cntrst-img}$. It can be considered as a special case where (1) the visual representation is extracted for a single global box that covers the whole image, (2) the corresponding text from the Internet describes the full image, and (3) negative samples are the text descriptions associated with other images. Finally, our overall loss function is given by

$$L = L_{cntrst} + L_{dist} + L_{cntrst-img}. \tag{3.5}$$

Zero-shot inference. Once pretrained, our visual encoder can be directly applied to region reasoning tasks. For example, given region proposals from RPN, region representations extracted from our visual encoder can be used to match the embeddings of target object concepts, and thus recognize the concepts within local image regions, thereby enabling zero-shot inference for object detection.

Transfer Learning for Object Detection

Our pretraining leverages region-text alignment created by the teacher model. Such alignment does not require human efforts, yet is not very accurate. When strong supervision for image regions is available (*e.g.*, the human-annotated detection labels), our visual encoder can be further fine-tuned by replacing the region descriptions with human annotations, as shown in Panel 3 of Fig. 3.2.

Specifically, we transfer our pretrained visual encoder to object detectors by initializing their visual backbones. To detect image objects, same as our pretraining, we use off-the-shelf RPN to localize object regions and recognize these regions by matching their visual region representation with the semantic embeddings of target object classes (*e.g.*, the object classes in detection dataset).

3.3 Key Experiments

Our main results are reported on transfer learning of our model for open-vocabulary object detection. Further, we evaluate our model on fully supervised object detection, as well as the zero-shot inference for object detection.

Datasets. For pretraining, we consider Conceptual Caption dataset (CC3M) Sharma et al. (2018) with 3 millions of image-text pairs from the web. We also use a smaller dataset COCO Caption (COCO Cap) Chen et al. (2015) when conducting ablation studies. COCO Cap contains 118k images, each associated with 5 human annotated captions. The parser from Jiayuan and Seito (2018) is adopted to extract triplets from captions in COCO Cap/CC3M dataset. Object concepts whose frequency are lower than 100 are discarded, leading to 4764/6790 concepts on COCO Cap/CC3M.

For transfer learning of open-vocabulary object detection, we train detectors with base categories of COCO detection dataset Lin et al. (2014) and LVIS dataset (v1) Gupta et al. (2019), respectively. On COCO, We follow the data split of Bansal et al. (2018) with 48 base categories and 17 novel categories which are subsets of COCO object classes. We use the processed data from Zareian et al. (2021) with 107,761 training images and 4,836 test images. On LVIS, following Gu et al. (2021), we use the training/validation images for training/evaluation and adopt the category split with 866 base categories (common and frequent) and 337 novel categories (rare).

Evaluation protocol and metrics. We evaluate object detection performance on COCO and LVIS for both transfer learning and zero-shot inference. The standard object detection metrics are used, including Average Precision (AP) and AP50 (AP at an intersection over union of 0.5).

Implementation details. *During pretraining*, the default student model and teacher model were ResNet50 He et al. (2016) from pretrained CLIP. RPN used in pretraining was trained with the base categories of LVIS dataset. Our default model was pretrained on CC3M dataset with the concepts parsed from COCO Cap. SGD was used with the batch size 96, initial learning rate 0.002, maximum iteration of 600k, and 100 regions per image. The temperature τ was 0.01.

For transfer learning of object detection, our detectors were developed on Detectron2 Wu et al. (2019) using Faster RCNN Ren et al. (2015a) (ResNet50-C4). RPN used in transfer learning was trained by the base categories of target dataset (*e.g.*, the transfer learning on COCO used the RPN trained on COCO). SGD was used with batch size 16, initial learning rate 0.002, and 1x schedule. Moreover, we applied

class-wise weighted cross-entropy loss. (1) For base categories, we used focal scaling with the weight for a base category as $(1-p^b)^\gamma$, where p^b is probability after softmax for this base category and $\gamma = 0.5/0.0$ on COCO/LVIS. Empirically, focal scaling helps to alleviate the forgetting of previously learned object concepts in pretraining, and thus is beneficial for novel categories. (2) For background category, we used a fixed all-zero embedding and a predefined weight (0.2/0.8 on COCO/LVIS) to background regions following Zareian et al. (2021).

For zero-shot inference of object detection, RPN was the same as pretraining stage and NMS threshold was set to 0.9. Inspired by Singh et al. (2018); Zhou et al. (2021), we fused RPN objectness scores and category confidence scores by geometry mean. Empirically, fusing RPN scores significantly improves zero-shot results.

Transfer to Open-Vocabulary Object Detection

Setup. We evaluate our models on two benchmarks for open-vocabulary object detection, including COCO and LVIS. On COCO, we report AP50 and follow the evaluation settings in Zareian et al. (2021): (1) only predicting and evaluating novel categories (Novel), (2) only predicting and evaluating base categories (Base), (3) a generalized setting that predicts and evaluates all categories (Generalized). On LVIS, we follow the benchmark of Gu et al. (2021) where the rare objects are defined as novel categories. We report AP for novel categories (APr), base categories (APc, APf) and all categories (mAP), respectively. The detectors are trained by base categories and evaluated on base and novel categories (*e.g.*, 48/866 base categories and 17/337 novel categories on COCO/LVIS). To compare with ViLD Gu et al. (2021), all experiments on LVIS additionally consider mask annotation.

Baselines. We consider several strong baselines:

- **Zero-shot object detectors** (SB Bansal et al. (2018), DELO Zhu et al. (2020), PL Rahman et al. (2020a)): Zero-shot object detection is the closest area to open-vocabulary object detection. These detectors usually rely on the pretrained word embeddings of object classes for generalization to novel categories.

- **Open-vocabulary object detectors** (OVR Zareian et al. (2021), ViLD Gu et al. (2021)): These detectors leverage pretrained vision-language models that have learned a large vocabulary from image-text pairs. OVR is our close competitor in the sense that we both pretrain visual encoders and use them as the detector

initialization. ViLD is a recent work that focuses on detector training by distilling visual features of a pretrained CLIP. ViLD specially uses data augmentation of large-scale jittering (LSJ) Ghiasi et al. (2021) with 16x training time.

- **Fully supervised detectors**: On COCO, we include the supervised baseline from OVR which is a Faster RCNN Ren et al. (2015a) trained by the base categories with 1x schedule. On LVIS, we include the supervised baseline from ViLD which is a Mask RCNN He et al. (2017) trained by base and novel categories with special data augmentation as ViLD. We additionally report a Mask RCNN trained in standard 1x schedule from Detectron2 Wu et al. (2019).

- **Our detector variants**: We consider initializing our detector with different visual encoders, including CLIP and our model pretrained on COCO Cap.

Results. In Table 3.1 and Table 3.2, we show the detection results on COCO and LVIS datasets, respectively.

On COCO dataset, initialized by our pretrained backbone, our detector significantly outperforms previous method OVR Zareian et al. (2021) on all metrics (*e.g.*, 31.4 vs. 22.8 on novel categories). Compared with the CLIP backbone from which we start our region-based pretraining, our model brings a remarkable gain across all metrics, particularly +17.2 AP50 on novel categories. When compared with ViLD, a recent SoTA method with sophisticated training strategy, our model is still comparable on Base and All, while substantially better on Novel (*e.g.*, 31.4 vs. 27.6) which is the main focus in open-vocabulary detection. On LVIS dataset, with comparable backbone size (RN50x4-C4 of ours: 83.4M, RN152-FPN of ViLD: 84.1M), our detector outperforms ViLD by a large margin (*e.g.*, +2.2 APr and +3.6 mAP). Note that these superior detection results on COCO and LVIS are achieved by using a single pretrained backbone, with standard data augmentation and 1x training schedule. These results suggest that our region-based vision-language pretraining has learned better alignment between image regions and object concepts, and thus facilitates open-vocabulary object detection.

Zero-shot Inference for Object Detection

Moving forward, we explore directly using RegionCLIP for zero-shot detection without any object annotations.

Visual Encoder Pretraining			Detector Training		COCO				
					Novel	Base	Generalized (17+48)		
Method	Dataset	Backbone	Method	Backbone	(17)	(48)	Novel	Base	All
Cls-ResNet He et al. (2016)	ImageNet	RN50	FR-CNN Ren et al. (2015a)	RN50-C4	-	54.5	-	-	-
Cls-IncRN Szegedy et al. (2017)	ImageNet	IncRNv2	SB Bansal et al. (2018)	IncRNv2	0.70	29.7	0.31	29.2	24.9
Cls-DarkNet Redmon et al. (2016)	ImageNet	DarkNet19	DELO Zhu et al. (2020)	DarkNet19	7.60	14.0	3.41	13.8	13.0
Cls-ResNet He et al. (2016)	ImageNet	RN50	PL Rahman et al. (2020a)	RN50-FPN	10.0	36.8	4.12	35.9	27.9
OVR Zareian et al. (2021)	COCO Cap	RN50	OVR Zareian et al. (2021)	RN50-C4	27.5	46.8	22.8	46.0	39.9
OVR Zareian et al. (2021)	CC3M	RN50	OVR Zareian et al. (2021)	RN50-C4	16.7	43.0	-	-	34.3
CLIP Radford et al. (2021)	CLIP400M	ViT-B/32	ViLD* Gu et al. (2021)	RN50-FPN	-	-	27.6	**59.5**	**51.3**
CLIP Radford et al. (2021)	CLIP400M	RN50	Ours	RN50-C4	22.5	53.1	14.2	52.8	42.7
Ours	COCO Cap	RN50	Ours	RN50-C4	30.8	55.2	26.8	54.8	47.5
Ours	CC3M	RN50	Ours	RN50-C4	**35.2**	**57.6**	**31.4**	57.1	50.4
Ours	CC3M	RN50x4	Ours	RN50x4-C4	**43.3**	**61.9**	**39.3**	**61.6**	**55.7**

Table 3.1: Open-vocabulary object detection results on COCO dataset. Initialized by our pretrained visual encoder, our detector outperforms previous works on all metrics by a remarkable margin, and outperforms the recent work ViLD* on novel categories. ViLD* trains the detector with data augmentation of large-scale jittering (LSJ) Ghiasi et al. (2021) and a much longer training schedule (16x). Notations: Cls denotes the image classification pretraining on ImageNet Deng et al. (2009), RN50 means ResNet50, IncRNv2 is Inception-ResNet-V2.

Visual Encoder Pretraining			Detector Training				LVIS			
Method	Dataset	Backbone	Method	Backbone	Training Strategy	Supervision	APr	APc	APf	mAP
-	-	-	Mask RCNN He et al. (2017)	RN50-FPN	16x+LSJ Ghiasi et al. (2021)	Base+Novel	13.0	26.7	**37.4**	28.5
Cls-ResNet He et al. (2016)	ImageNet	RN50	Mask RCNN He et al. (2017)	RN50-C4	1x+Standard	Base+Novel	11.9	22.0	29.7	23.3
CLIP Radford et al. (2021)	CLIP400M	ViT-B/32	ViLD* Gu et al. (2021)	RN50-FPN	16x+LSJ Ghiasi et al. (2021)	Base	16.7	26.5	34.2	27.8
Ours	CC3M	RN50	Ours	RN50-C4	1x+Standard	Base	17.1	27.4	34.0	28.2
CLIP Radford et al. (2021)	CLIP400M	ViT-B/32	ViLD* Gu et al. (2021)	RN152-FPN	16x+LSJ Ghiasi et al. (2021)	Base	19.8	27.1	34.5	28.7
Ours	CC3M	RN50x4	Ours	RN50x4-C4	1x+Standard	Base	**22.0**	**32.1**	36.9	**32.3**

Table 3.2: Open-vocabulary object detection results on LVIS dataset. Without sophisticated training strategy, our detector still outperforms ViLD* on most metrics. Using same training strategy, our open-vocabulary detector beats the fully-supervised Mask RCNN for all metrics.

Setup. The pretrained vision-language models are directly used to recognize image regions. We use the same evaluation datasets and metrics as the experiments in transfer learning (All AP50 for COCO, mAP for LVIS) [1] . We consider two settings: (1) Ground-truth (GT) bounding boxes are used as region proposals. This oracle setting aims at evaluating the recognition performance by eliminating the localization error; (2) The region proposals come from RPN used in pretraining. The performance is thus impacted by both the quality of localization and accuracy of recognition.

Baselines. We consider two baselines: (1) OVR Zareian et al. (2021) pretrains a visual backbone on image-text pairs of COCO Cap which has close object concepts as COCO detection dataset. We evaluate the pretrained model provided in their code base. (2)

[1]The breakdown metrics (*e.g.*, Novel and Base) are omitted in zero-shot inference since no detection annotations are used.

Method	Visual Encoder Pretraining Dataset	Backbone	Region Proposals	COCO All	LVIS mAP
OVR Zareian et al. (2021)	COCO Cap	RN50	GT	44.5	-
CLIP Radford et al. (2021)	CLIP400M	RN50	GT	58.3	42.2
Ours	CC3M	RN50	GT	61.4	44.4
Ours	CC3M	RN50x4	GT	**65.5**	**50.7**
OVR Zareian et al. (2021)	COCO Cap	RN50	RPN	19.6	-
CLIP Radford et al. (2021)	CLIP400M	RN50	RPN	25.5	9.2
Ours	CC3M	RN50	RPN	26.8	9.6
Ours	CC3M	RN50x4	RPN	**29.6**	**11.3**

Table 3.3: Zero-shot inference with ground-truth (GT) boxes or RPN boxes on COCO and LVIS datasets. All models use RoIAlign to extract visual representation of proposed regions. Our pretrained models beat baselines by a clear margin across datasets.

CLIP Radford et al. (2021) is pretrained on 400M image-text pairs. Both OVR and CLIP consider image-text pairs for pretraining, same as our RegionCLIP.

Results. Table 3.3 summarizes the results. With GT boxes, our pretrained model outperforms CLIP baseline by a clear margin across datasets (*e.g.*, 61.4 vs. 58.3 All AP50 on COCO, 44.4 vs. 42.2 mAP on LVIS). When compared with OVR, our model demonstrates a much larger margin (*e.g.*, 61.4 vs. 44.5 All AP50 on COCO), not to mention that OVR is pretrained on the same dataset as evaluation. When using RPN proposals, our model still clearly outperforms CLIP and OVR (*e.g.*, 26.8 vs. 19.6 & 25.5 on COCO, 9.6 vs. 9.2 on LVIS). Note that using GT boxes better characterizes the recognition performance of a pretrained model than using RPN, since RPN injects additional localization errors. These results suggest that our pretraining with region-text alignment improves the recognition of image regions. With RN50x4 architecture as the backbones of teacher and student models, the zero-shot inference performance is further improved across datasets and settings (*e.g.*, +6.3 mAP on LVIS with GT, +2.8 All AP50 on COCO with RPN).

Visualization. Fig. 3.3 visualizes the results of zero-shot inference with GT boxes on COCO dataset. Our model predicts more reasonable object categories than CLIP (*e.g.*, the blue regions in 1st and 2nd columns are correctly predicted as "umbrella" and "person" by our model). These results suggest that our proposed region-based vision-language pretraining can help to recognize image regions precisely.

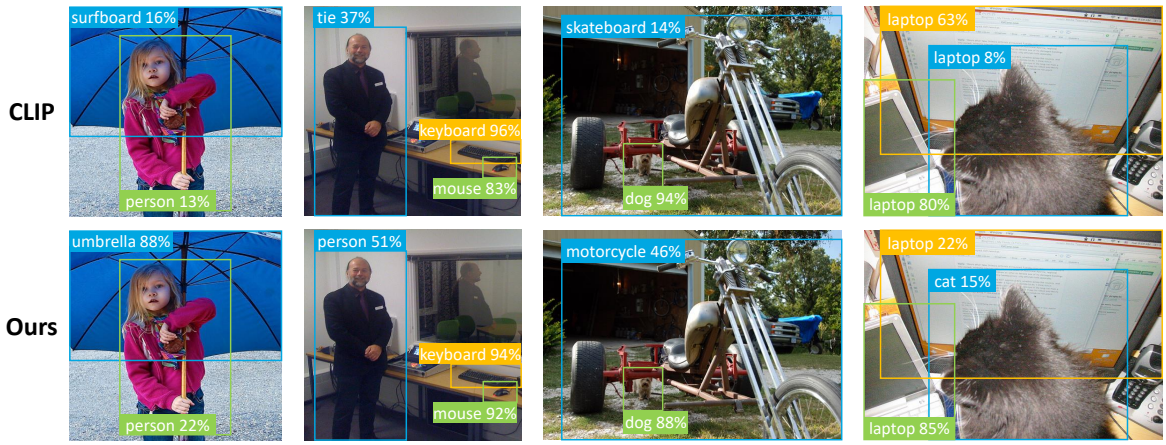


Figure 3.3: Visualization of zero-shot inference on COCO dataset with *ground-truth boxes*. Without finetuning, the pretrained models (top: CLIP, bottom: Ours) are directly used to recognize image regions into the categories in COCO.

3.4 Conclusion

In this chapter, we proposed RegionCLIP — a novel region-based vision-language pretraining method that learns to match image regions and their descriptions. The key innovation is a scalable approach to associate region-text pairs without using human annotation. By learning from such region-level alignment, the pretrained model in my work established new state of the art when transferred to open-vocabulary object detection on COCO and LVIS datasets. Moreover, the same pretrained model demonstrated promising results on zero-shot inference for object detection. We believe my work provides a solid step towards region representation learning, and we hope that my work can shed light on vision-language pretraining.

4 LEARNING OBJECT RELATIONSHIP

Chapter 3 demonstrated my work of learning object concepts from image-text pairs. In this chapter, we ask a further question: *can models learn to detect object relationships beyond individual object concepts from image-text pairs?* Fig. 4.1 illustrates an example of such relationships ("man *drive* boat"). We focus on learning scene graph generation (SGG) from the training data of image-text pairs, as shown in Fig. 4.1. A scene graph is a symbolic and graphical representation of an image, with each graph node as a localized object and each edge as a relationship (*e.g.*, a predicate) between a pair of objects. During training, we use image-text pairs as the training data, without any human-annotated scene graphs. During inference, our model takes an image and its detected objects as inputs and outputs a scene graph.

Most previous scene graph methods Xu et al. (2017); Li et al. (2017); Zellers et al. (2018); Li et al. (2018b); Yang et al. (2018a); Chen et al. (2019a); Tang et al. (2019a, 2020a); Zhang et al. (2019) follow a fully supervised approach, relying on human annotations of object bounding boxes, object categories and their relationships. These annotations are very costly and difficult to scale. Recently, Zareian *et al*. Zareian et al. (2020) considered weakly supervised learning of scene graphs from image-level labels of unlocalized scene graphs. Nonetheless, learning scene graphs from images and their text descriptions remains unexplored. Table 4.1 shows the comparison between our language supervised setting and previous settings (fully or weakly supervised). Our setting provides a new opportunity of learning structured visual knowledge from natural language supervision.

In this chapter, we propose the first method that learns object relationships from image-text pairs so that the trained model can output a scene graph for the input image. A major challenge of learning scene graphs from image-sentence pairs is the missing link between many candidate image regions and a few concepts (*e.g.*nouns and predicates) parsed from an image caption. To this end, we propose to leverage off-the-shelf object detectors, capable of identifying and localizing object instances from hundreds of common categories. Our key idea is that the object labels of detected image regions can be further matched to sentence concepts, and thus provide "pseudo" triplet labels (subject-predicate-object) for learning scene graphs, thereby bridging the gap between region-concept pairs. Our hypothesis is that these "pseudo" labels, coupled with a large-scale dataset, can be used for training a deep model to detect

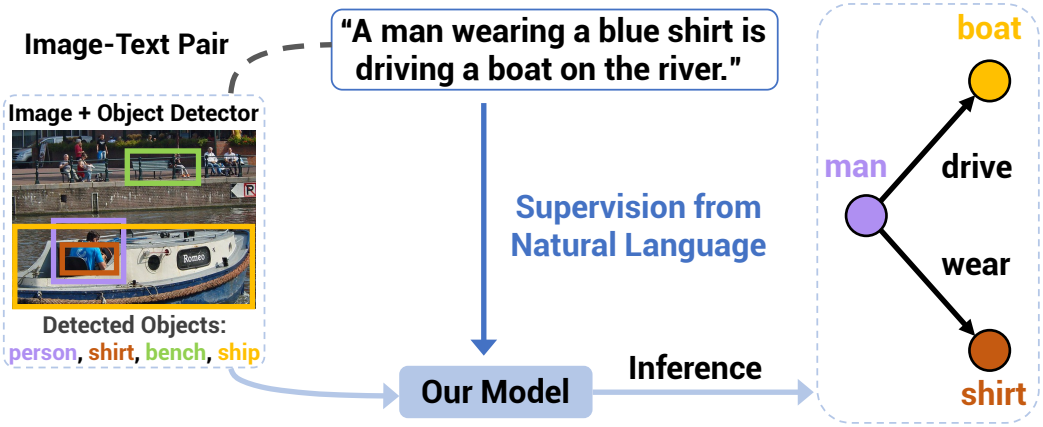


Figure 4.1: Our setting: Our goal is learning to generate localized scene graphs from image-text pairs. Once trained, our model takes an image and its detected objects as inputs and outputs the image scene graph.

Scene Graph Generation Settings	Required Annotation during Training		
	Image Description	Object&Predicate Category Labels	Object Boxes
Fully Supervised Xu et al. (2017)		✓	✓
Weakly Supervised Zareian et al. (2020)		✓	
Language Supervised (ours)	✓		

Table 4.1: Our language supervised setting vs. fully and weakly supervised settings. Our method learns from only image-text pairs to generate localized image scene graphs, without using human-annotated scene graphs (object location, object & predicate category labels).

scene graph of an input image.

Specifically, we develop a Transformer-based model for learning to generate scene graphs supervised by image-sentence pairs, inspired by the recent success of vision-language pretraining Chen et al. (2020b); Li et al. (2020b); Zhou et al. (2020); Lu et al. (2020); Su et al. (2020); Tan and Bansal (2019); Lu et al. (2019). Our model takes inputs of visual features from a pair of detected object regions, text embeddings of their predicted categorical labels, and contextual features from other object regions, all provided by an off-the-shelf detector Ren et al. (2015a). Our model then learns to recognize the visual relationship between the input object pair, represented as a localized subject-predicate-object (SPO) triplet. A scene graph can thus be generated by enumerating all pairs from a small set of detected objects. During training, our model learns from only image-sentence pairs using "pseudo" labels produced by matching the detected object labels to the parsed sentence concepts. During inference,

our model generates a scene graph given an input image with its detection results.

This chapter is organized as follows. In Section 4.1, I summarize our contributions. In Section 4.2 and Section 4.3, I present our method and key experiments, respectively. Finally, I conclude the chapter in Section 4.4.

4.1 Contributions

The contributions of this chapter are summarized into four folds:

- We propose one of the first methods of learning to generate scene graphs from image-text pairs.
- The key innovation is using off-the-shelf object detector and language parser to create pseudo triplet labels (subject-predicate-object), by matching image regions to text tokens.
- Our model trained by image-text pairs produces high-quality scene graphs, outperforming the latest model trained by human-annotated scene graphs.
- We further present the first results for open-set scene graph generation. The promising results suggest that structured visual representation with open-set concepts can be learned from natural language.

This work was a collaboration with Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. The work has been published by ICCV 2021 with title as "Learning to Generate Scene Graph from Natural Language Supervision" Zhong et al. (2021).

4.2 Method

With a large collection of paired images $\{I\}$ and captions $\{S\}$, our goal is learning to detect an image scene graph $G = (V, E)$ from an input image I. G is a directed graph with nodes V and edges E. Each node $v_i \in V$ denotes a localized object in I, represented by its bounding box b_i and object label o_i within a vocabulary C_o^g. Each edge $e_{ij} \in E$ denotes a predicate (*e.g.*"drive") from a vocabulary C_p^g pointing from node v_i to node v_j. $T_{ij} = (v_i, e_{ij}, v_j)$ defines a triplet of subject-predict-object (SPO). Scene graph generation is thus a challenging problem of structured output prediction.

Similar to previous SGG methods Xu et al. (2017); Zhang et al. (2017a); Tang et al. (2019a); Zellers et al. (2018); Zareian et al. (2020), we assume a set of object regions $R = \{r_n\}$ provided by an detector. Each region $r_n = (\bar{b}_n, \bar{o}_n)$ consists of a bounding box $\bar{b}_n$ and a predicted object category $\bar{o}_n$ from a vocabulary C_o^d given by the detector. r_n thus defines a candidate node of the target scene graph G. It is worth noting that the vocabulary of the detector C_o^d is different from the vocabulary of the scene graph C_o^g (*i.e.*$C_o^d \neq C_o^g$). With object regions $R = \{r_n\}$, SGG is reduced to classify r_n into object categories $(C_o^g \cup \{\text{background}\})$, and infer the predicate label $(C_p^g \cup \{\text{background}\})$ between each subject-object region pair (r_k, r_l). A main innovation of our model is to learn from only image-text pairs for SGG, without the need of ground-truth object labels nor their relationships.

Learning from Language Supervision. Our key idea is to extract SPO triplets from an image caption, and match these triplets to object categories of image regions given by the detector, thereby creating "pseudo" labels for these regions and their relationships. Specifically, we adopt a language parser Jiayuan and Seito (2018); Schuster et al. (2015) to extract a set of triplets $\{T'\}$ from the caption S. We further link object region pairs $\{r_k, r_l\}$ in the image I provided by the detector to the parsed sentence triplets T'. This is done by using WordNet Miller (1995) to match detected object categories $\bar{o}_k$ and $\bar{o}_l$ from every region pair to the subject and object in each T', respectively. If matched, the sentence triplet T' will define a "pseudo" label for the region pair (r_k, r_l) (subject, object) and their relationship e_{kl} (predicate). These "pseudo" labels can then be used to train our model.

Model overview. Our model, inspired by recent work in vision-language pretraining Chen et al. (2020b); Li et al. (2020b); Zhou et al. (2020); Lu et al. (2020); Su et al. (2020); Tan and Bansal (2019); Lu et al. (2019), seeks to label the SPO triplet given a pair of regions. Specifically, we design a Transformer-based model with its inputs as a region pair (r_k, r_l) and the contextual features from other regions $\{r_n\} - \{r_k, r_l\}$. Our model then predicts the category labels (o_k, e_{kl}, o_l) of a SPO triplet T_{kl} for the input region pair (r_k, r_l). During training, our model is supervised by the "pseudo" labels T' parsed from caption S. During inference, our model takes inputs of the image I and its detection results $R = \{r_n\}$, labels every region pair (r_k, r_l), and aggregates the SPO triplets into a full scene graph. Fig. 4.2 illustrates our model.

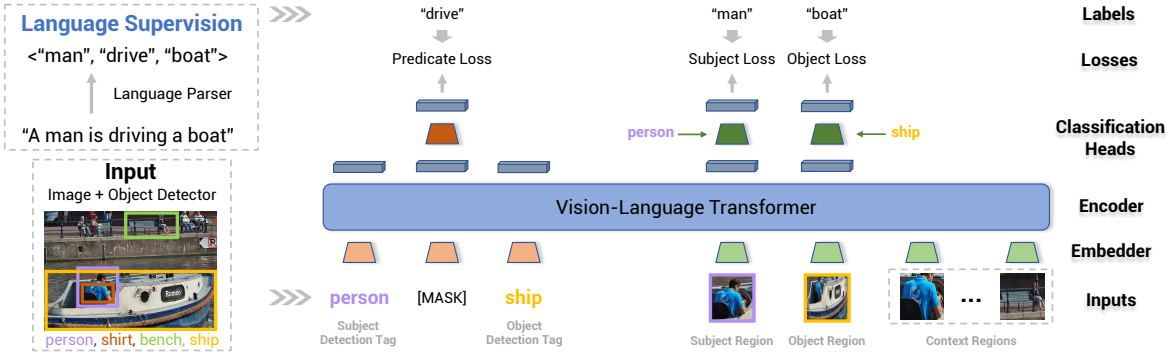


Figure 4.2: Overview of our proposed model for language supervised scene graph generation. Given an image, an object detector is first applied with the detected objects as the inputs to our model. Our model further embeds the detected region features and textual object categories (e.g., the tags of a pair of subject-object, the MASK representing the predicate) into token embeddings, followed by a multi-layer Transformer encoder. Finally, our model predicts the labels of the subject region, the object region and the predicate.

Triplet Transformer

Our proposed Triplet Transformer is a triplet labeling model based on an input region pair and its contextual features. Specifically, for each region $r_n = (\bar{b}_n, \bar{o}_n)$, we denote its visual, positional, and textual features as $\mathbf{x}^r_n$, $\mathbf{x}^p_n$, and $\mathbf{x}^o_n$, respectively. $\mathbf{x}^r_n$ is the visual feature (ROI) pooled from the region $\bar{b}_n$. $\mathbf{x}^p_n$ is a feature encoding the position of the bounding box, *i.e.*, a 7-D vector with the normalized top/left/bottom/right coordinates, width, height and area for the region box $\bar{b}_n$. $\mathbf{x}^o_n$ is the word embedding of the region object label $\bar{o}_n$. Given an input region pair (r_k, r_l) and all other detected regions, our model builds a composition function $f = g \circ h$ to predict the labels (o_k, e_{kl}, o_l) of a SPO triplet, given by

$$o_k, e_{kl}, o_l = g \circ h \left(\underbrace{\mathbf{x}^o_k, \mathbf{x}^o_l}_{\text{Textual Embedder}}; \underbrace{\mathbf{x}^r_k, \mathbf{x}^r_l, \mathbf{x}^p_k, \mathbf{x}^p_l}_{\text{Visual Embedder}}; \overbrace{\{\mathbf{x}^r_u, \mathbf{x}^p_u\}_{u \neq k,l}}^{\text{Contextual Features}} \right)$$

where u indexes all regions except r_k and r_l.

Our model thus consists of: (1) a visual embedder that encodes visual and positional region features; (2) a textual embedder that embeds textual region features (from object labels); (3) a multi-layer Transformer h that conducts message passing among the input visual and textual embeddings; and (4) classification heads g that

predict the labels of a triplet. We now present details for each component.

Visual Embedder. Our visual embedder transforms visual and positional features ($\mathbf{x}^r_n$ and $\mathbf{x}^p_n$) of region r_n into an embedding $\mathbf{v}_n$, where n indexes all region features including k (subject), l (object) and u (context). This is given by

$$\mathbf{v}_n = \text{LN}(\text{LN}(\mathbf{W}_r \mathbf{x}^r_n) + \text{LN}(\mathbf{W}_p \mathbf{x}^p_n) + \mathbf{e}^t_n), \tag{4.1}$$

where $\mathbf{W}_r$ and $\mathbf{W}_p$ are trainable weights that project the features into the same dimension d. $\mathbf{e}^t_n \in \mathbb{R}^d$ is the type embedding of a region (subject vs. object vs. context). LN denotes Layer Normalization Ba et al. (2016).

Textual Embedder. Our textual embedder accepts two inputs: (1) the word embeddings $\mathbf{x}^o_k$ and $\mathbf{x}^o_l$ of region labels for subject and object region, respectively; and (2) the word embedding of a special word "MASK", denoted as $\mathbf{x}^o_p$, representing the missing predicate. The embedder encodes the input word embedding and the positional embedding into a textual embedding $\mathbf{t}_m$, given by

$$\mathbf{t}_m = \text{LN}(\mathbf{W}_e \mathbf{x}^o_m + \mathbf{e}^p_m), \tag{4.2}$$

where m indexes k (subject), p (predicate) or l (object). $\mathbf{e}^p_m \in \mathbb{R}^d$ is the positional embedding Devlin et al. (2019) of the current token. $\mathbf{W}_e$ represents the trainable weights projecting the word embedding into the dimension of d.

Transformer Encoder. The visual and textual embeddings ($\mathbf{v}_n$ and $\mathbf{t}_m$) are further fed into a multi-layer Transformer encoder Vaswani et al. (2017a). This encoder uses multi-head self-attention, coupled with multilayer perceptron (MLP) and layer normalization, to output a contextualized embedding ($\hat{\mathbf{v}}_n \in \mathbb{R}^d$ or $\hat{\mathbf{t}}_m \in \mathbb{R}^d$) for each input $\mathbf{v}_n$ or $\mathbf{t}_m$. This Transformer encoder can be considered as conducting message passing across all input tokens. Among all the outputs, the embeddings corresponding to the subject, predicate, object tokens will be further used for triplet label prediction, as shown in Fig. 3.2. For a region pair (r_k, r_l), the embeddings $\hat{\mathbf{v}}_k$ / $\hat{\mathbf{t}}_k$ correspond to the visual / textual feature of the subject region (*i.e.*the first input region), the predicate embedding $\hat{\mathbf{t}}_p$ is from the special word "MASK", and the embeddings $\hat{\mathbf{v}}_l$ / $\hat{\mathbf{t}}_l$ represent the visual / textual feature of the object region (*i.e.*the second input region).

Classification Heads. Our model further fuses the encoder outputs, and predicts labels of a SPO triplet (subject-predicate-object) for the input region pair (r_k, r_l). The

feature fusion is given by

$$\begin{aligned} \mathbf{s} &= \hat{\mathbf{v}}_k + \mathbf{W}_v \mathbf{x}^o_k, \quad \mathbf{o} = \hat{\mathbf{v}}_l + \mathbf{W}_v \mathbf{x}^o_l, \\ \mathbf{p} &= \hat{\mathbf{t}}_p + \mathbf{W}_{ts} \hat{\mathbf{t}}_k + \mathbf{W}_{to} \hat{\mathbf{t}}_l + \mathbf{W}_{vs} \hat{\mathbf{v}}_k + \mathbf{W}_{vo} \hat{\mathbf{v}}_l, \end{aligned} \tag{4.3}$$

where $\mathbf{W}_v$, $\mathbf{W}_{ts}$, $\mathbf{W}_{to}$, $\mathbf{W}_{vs}$, $\mathbf{W}_{vo}$ are learnable weights. The outputs $\mathbf{s} \in \mathbb{R}^d$, $\mathbf{o} \in \mathbb{R}^d$, $\mathbf{p} \in \mathbb{R}^d$ are further used to classify subject, predicate, and object labels, respectively. This is done using a two-layer MLP followed by softmax.

Learning from Language Supervision

Our key innovation is the use of image captions as the only supervisory signal for training our model. This is done by constructing "pseudo" labels of triplets from image captions. Concretely, we first parse a caption into a set of SPO triplets. Each triplet is further matched to every pair of regions, by comparing subject and object tokens in the sentence triplet to the predicted categories of a region pair. Our model is then trained on the matched pairs of regions to predict their corresponding sentence triplets. We point out that our approach of learning from image-sentence pairs can be easily adapted by different SGG models.

Closed-Set vs. Open-Set. In this paper, we primarily consider a closed-set setting — the vocabulary of the subject, predicate, and object during evaluation is known in prior. In this setting, our learning is focused on the concepts of interest and our model only considers sentence triplets within the vocabulary. Nonetheless, our method does support the open-set setting, where there are no limits on the vocabulary. In this case, our model learns from all frequently appearing subject, predicate, and object tokens in the captions. Additional matching step is needed at inference time to identify concepts in the target vocabulary. We will explore this setting in our experiment.

Triplet Parsing and Filtering. We use an off-the-shelf rule-based language parser Jiayuan and Seito (2018) based on Schuster *et al*.Schuster et al. (2015) to parse the triplets in the image captions. After parsing, the triplets with the lemmatized words for subject, predicate and object are obtained. We further perform an optional filtering step on the initial collection of triplets. For the closed-set setting, we only keep concepts that can be matched to the categories in the target vocabulary. Two concepts are matched if (1) there is overlapping between their synsets, lemmas or hypernyms in WordNet Miller (1995) (*e.g.*"tortoise" $\rightarrow$ "animal"), or (2) if their root forms can

be matched (*e.g.*"baseball player" $\rightarrow$ "player").

Pseudo Label Assignment. With the filtered triplets, our next step is to match sentence triplets to pairs of regions provided by the object detector. This is done by a greedy matching between every triplet from the caption and each region pair from the image. Specifically, we match the corresponding subject and object tokens between a triplet and a region pair, again using a token's synsets, the synsets' lemmas and hypernyms in WordNet Miller (1995) and its root form. If multiple triplets are matched to the same region pair, we randomly select one of them. We also filter out region pairs that does not overlap and far away from each other, following Zellers et al. (2018), as these pairs are less likely to contain a relationship. Once matched, the triplet is considered as the pseudo label of the region pair for training our model.

Model Training. Our model is trained by predicting the pseudo labels of the region pairs. We apply a multi-class cross-entropy loss for the subject, predicate, and object, respectively. Our final loss function is given by

$$L = \lambda_s L_s + \lambda_p L_p + \lambda_o L_o \tag{4.4}$$

where L_s, L_p, and L_o is the loss for subject, predicate, and object respectively. And λ_s, λ_p, and λ_o are their corresponding loss weights. We set $\lambda_s = \lambda_o = 0.5$ and $\lambda_p = 1$ following previous work Zellers et al. (2018); Zareian et al. (2020).

Weighted Loss. One challenge for learning is the domain gap between (a) image-sentence pairs used for training and (b) images and their target scene graphs during inference. For example, the distributions of concepts might be quite different in image-sentence pairs vs. image scene graphs. In the closed-set setting, we might have an estimated frequency of the concepts on scene graphs. In this case, we apply a weighted loss during training, where the weight for each category is set to the ratio between the frequency of the token in image-sentence pairs and the estimated frequency of the matched tokens in scene graphs. If a category is not matched to any target category, no loss weight will be applied. This weighted loss function only requires an estimated frequency of concepts on the target dataset, and can be considered as a simple approach for domain adaption.

Model Inference. Once trained, our model takes a region pair and its contextual features, and predicts a SPO triplet. To obtain a scene graph, we enumerate all possible region pairs and feed them into our model. The predicted probabilities are further averaged for each region and thus each region is predicted to single category. In the

open-set setting, an additional matching step is needed to infer the probability of target categories based on the predicted categories from image-sentence pairs. In this case, we apply the same matching step in our label assignment step.

Extension to Weakly and Fully Supervised Settings. Our model can be easily extended to weakly and fully supervised settings. In weakly supervised setting, we replace triplets parsed from captions with those from unlocalized scene graphs Zareian et al. (2020), and follow the same label assignment of our setting. For fully supervised setting, we simply replace our pseudo labels with ground-truth scene graph labels.

4.3 Key Experiments

In this section, I present our experiments and results. We start with our main results on learning SGG from image-sentence pairs. Further, we explore open-set SGG.

Datasets. To evaluate our model, we used the standard split Xu et al. (2017) of Visual Genome (VG) Krishna et al. (2017) (150 objects, 50 predicates, 75K/32K images for train/test). VG comes with human-annotated image captions and localized scene graphs, and is a widely used benchmark for SGG. For training, we considered image captions from VG, COCO Caption (COCO) Chen et al. (2015), and Conceptual Caption (CC) Sharma et al. (2018). COCO contains 123K images with each labeled by 5 human-annotated captions. We selected 106k images in COCO for training by filtering out images that exist in the test set of VG. CC contains 3.3M image-caption pairs automatically collected from alt-text enabled images on the web. For the closed-set setting where the target categories are known, we matched the parsed tokens from each dataset to target categories, and kept 148-52, 143-56, 148-64 object-predicate categories for VG, COCO and CC, respectively, leading to 673K/75K (triplets/images) on VG, 154K/64K on COCO, and 159K/145K on CC.

Evaluation Protocol and Metrics. For the majority of our experiments, we evaluate Scene Graph Detection (SGDet) following the protocol from Xu et al. (2017). SGDet captures both the localization and classification performance using metrics of Recall@K (R@K) Lu et al. (2016); Xu et al. (2017) and mean Recall@K (mR@K) Chen et al. (2019b); Tang et al. (2019a). R@K computes the recall between the top K predicted triplets and ground-truth ones. A predicted triplet is considered as correct only when all requirements are met: (1) the predicted triplet labels match one of the ground-truth triplet, (2) the detected subject-object regions match the ground-truth

subject-object regions with an IoU $\geqslant 0.5$, respectively. mR@K averages R@K across all predicate categories. We also included Scene Graph Classification (SGCls) and Predicate Classification (PredCls) in our experiment on fully SGG. Importantly, all experiments were conducted with graph constraint that limits each subject-object pair to have only one predicate prediction.

Implementation Details. We used a Faster R-CNN Ren et al. (2015a) detector pre-trained on OpenImages Kuznetsova et al. (2020), capable of detecting 601 object categories. We kept the top 36 objects per image and extracted the 1536-D region features from the detector. The object tags were represented by the 300-D GloVe embeddings Pennington et al. (2014). We adopted the Transformer implementation from UNITER Chen et al. (2020b) with 2 self-attention layers, 12 attention heads in each layer and hidden size $d = 768$. SGD optimizer was used in training with the image batch of 32, 16 sampled triplets per image, and the initial learning rate of 0.0032. We used the benchmark provided by Tang et al. (2020a) for evaluation.

Language Supervised Scene Graph Generation

Now I present our main results on learning to generate scene graphs with image-sentence pairs as only training data.

Setup and Baselines. We consider several baselines and variants of our model. A key feature of our model is the ability to learn from only image-sentence pairs.

- **VSPNet** Zareian et al. (2020) is designed for weakly supervised SGG and learns from unlocalized scene graph. As our close competitor, VSPNet takes the inputs of object proposals from the same OpenImage detector used by our model.
- **VSPNet†** further augments VSPNet with object box predictions from the detector. VSPNet† thus has the same input image regions as our model.
- **Ours+Weak** is our model trained using unlocalized scene graphs, same as the setting of VSPNet.
- **Ours+MotifNet** combines our pseudo label assignment with a supervised SGG model (MotifNet Zellers et al. (2018)). This model is thus trained using only image-sentence pairs.
- **Ours+Full** is our model trained with full supervision and using ground-truth scene graph labels. This should be considered as an upper bound of our model.

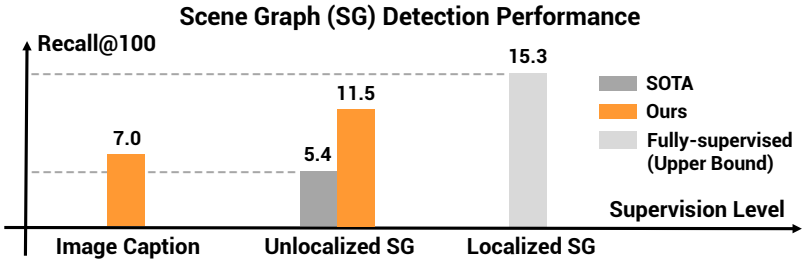


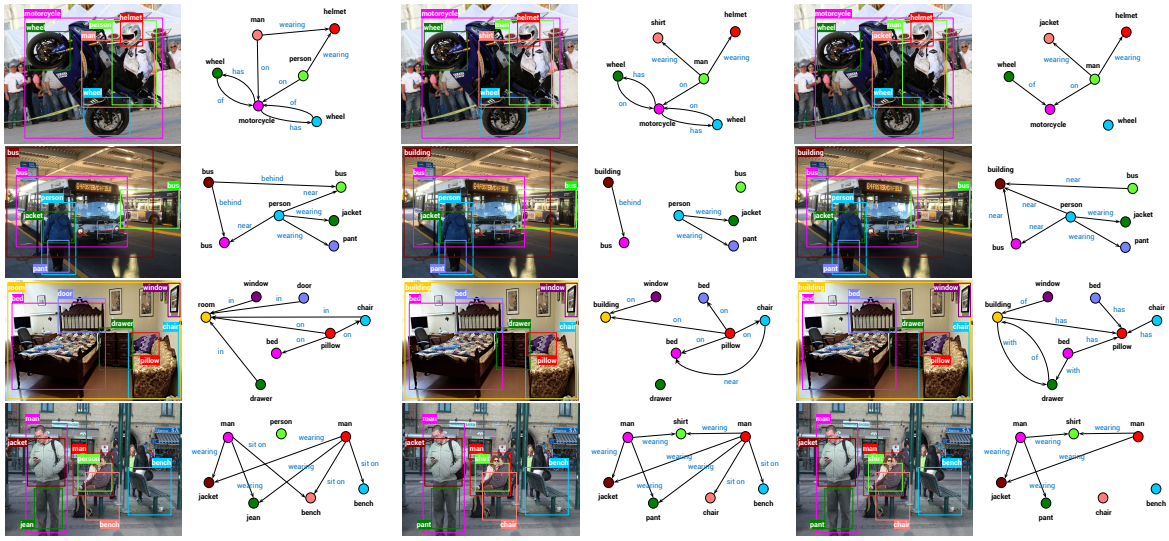
Figure 4.3: Main results: A comparison of results from our method and the state-of-the-art (SoTA) method VSPNet with varying levels of supervision.

Method	Training Setting					SGDet	
	Supervision	Level	Source	#Triplets	#Images	R@50	R@100
Ours+Full	Localized Scene Graph	Full	Visual Genome	406K	58K	**13.8**	**15.3**
VSPNet Zareian et al. (2020)	Unlocalized Scene Graph	Weak	Visual Genome	406K	58K	4.7	5.4
VSPNet†						6.7	7.4
Ours+Weak						**10.0**	**11.5**
Ours+MotifNet	Image Description	Weaker	CC + COCO	313K	210K	5.6	6.7
Ours						**5.9**	**7.0**

Table 4.2: Results of language supervised SGG. Different from all previous approaches, our model can learn from image-sentence pairs for SGG. With only image-sentence pairs as the supervisory signal, our model outperforms VSPNet — a latest method of weakly supervised SGG trained using human-annotated, unlocalized scene graphs.

Results. Fig. 4.3 presents our main results. Our model, trained by only image-text pairs, significantly outperforms the state-of-the-art method VSPNet by a relative margin of **30%**, despite that VSPNet is trained using human-annotated unlocalized scene graphs. With the same supervision as VSPNet, our model achieves a relative gain of **112%** in recall. These results provide **first** convincing evidence that high-quality scene graphs can be learned from only image-text pairs.

Table 4.2 shows additional detailed comparison. With image description (CC + COCO) as only supervision, our models (Ours/Ours+MotifNet) significantly outperform VSPNet trained using unlocalized scene graphs (7.0/6.7 vs. 5.4 R@100), despite that image-sentence pairs are much weaker supervisory signals. Our Transformer-based model also beats Ours+MotifNet, and performs on par with the improved version of VSPNet (VSPNet†) (7.0 vs. 7.4 R@100). When trained using unlocalized scene graphs, our model (Ours+Weak) again outperforms VSPNet variants by a large margin (11.5 vs. 5.4/7.4 R@100). These results suggest that our model can learn from only image-sentence pairs to detect scene graph in an image with high quality. Finally, there is a noticeable gap between Ours and Ours+Weak (7.0 vs. 11.5 R@100), and between Ours+Weak and Ours+Full (11.5 vs. 15.3 R@100), suggesting ample



(a). Trained by Localized Scene Graphs (b). Trained by Unlocalized Scene Graphs (c). Trained by Image Descriptions

Figure 4.4: Qualitative results of our models on VG test set for SGG. All models take the same detected regions and predict the scene graph labels. In each row, we show 3 identical images and the corresponding scene graphs generated from the models trained by different levels of supervision. The visualized relationships are picked from the top 30 predicted triplets.

room for future work.

Fig. 4.4 further visualizes the output scene graphs from our models, including Ours+Full (left), Ours+Weak (middle) and Ours (right) in Table 4.2. Our model trained by image-sentence pairs produces scene graphs with a comparable quality as those trained using strong supervision (*e.g.*"man-on-motorcycle" and "man-wearing-helmet" in the 1st row). Further, our models trained using scene graphs tend to predict a different set of predicate when compared to our model trained using image-sentence pairs. This is best illustrated in the 3rd row of Fig. 4.4 ("on" vs. "has"). We conjecture that this is caused by different distributions of predicates in scene graph and in image captions. Finally, it is worth noting that similar to many previous approaches, our models fall short when common sense reasoning is needed. This is shown in the 4th row of Fig. 4.4, where our models predict two man wearing the same jacket or shirt.

Comparison to LSWS Ye and Kovashka (2021). In addition, we compare our results to a concurrent work of LSWS Ye and Kovashka (2021). LSWS also learns to generate scene graph from image-sentence pairs using iterative visual grounding. Table 4.3 summarizes the comparison. When trained with the same level of supervision and the

Method	Training Setting		SGDet	
	Supervision	Source	R@50	R@100
LSWSYe and Kovashka (2021)	Unlocalized Scene Graph	Visual Genome	7.3	8.7
Ours			**10.0**	**11.5**
LSWSYe and Kovashka (2021)	Image Description	Visual Genome	3.9	4.0
Ours		Visual Genome	**9.2**	**10.3**
LSWSYe and Kovashka (2021)		COCO	3.3	3.7
Ours		COCO	**5.8**	**6.7**

Table 4.3: Comparison to the concurrent work of LSWS Ye and Kovashka (2021).

same dataset, our models constantly outperform LSWS by a large margin. For example, when trained using image-sentence pairs on COCO, our method achieves 5.8 R@50 and 6.7 R@100 vs. 3.3 R@50, and 3.7 R@100 from LSWS — a relative gain of at least 75%. When trained with unlocalized scene graph as the setting in VSPNet Zareian et al. (2020), our model also outperforms LSWS by a noticeable margin (+2.7 R@50 and +2.8 R@100).

Open-set Scene Graph Generation

Moving forward, we consider a challenging open-set setting for SGG, where the categories of target concepts (objects and predicates) are unknown during training. We believe this is the first result for open-set SGG.

Setup. In this experiment, our model is trained on COCO Caption and evaluated on VG. During training, we parsed concept categories from captions, remove the low-frequency categories, and formed a vocabulary of 4273 objects and 677 predicates. This vocabulary was then used to train our model. At inference time, we first generated scene graphs using our vocabulary, and then matched the detected categories in our vocabulary to target concepts on VG (150 objects and 50 predicates) for evaluation.

Results. Table 4.4 compares the results of our models trained in closed-set and open-set settings using the same COCO caption dataset. The model trained in open-set setting has slightly better recall (4.8 vs. 4.5 R@100). Our open-set results are also comparable to VSPNet (supervised by unlocalized scene graphs on VG in a closed-set setting). We hypothesis that the open-set setting allows the model to learn from more concepts and thus benefits SGG. To verify this hypothesis, we plot the output scene graph from our models trained on closed-set and open-set settings in Fig. 4.5. Compared to our closed-set model, our open-set model detects more concepts outside VG (*e.g.*"swinge", "mouse", "keyboard"). Our results suggest an exciting avenue of large-scale training of open-set SGG using image captioning dataset such as CC.

Model	#Objects	#Predicates	#Triplets	#Images	SGDet R@50	SGDet R@100
Ours	143	56	154K	64K	3.8	4.5
Ours	4273	677	758K	105K	4.1	4.8

Table 4.4: Results of open-set SGG. Evaluation is performed on VG with the vocabulary and model learned from COCO.

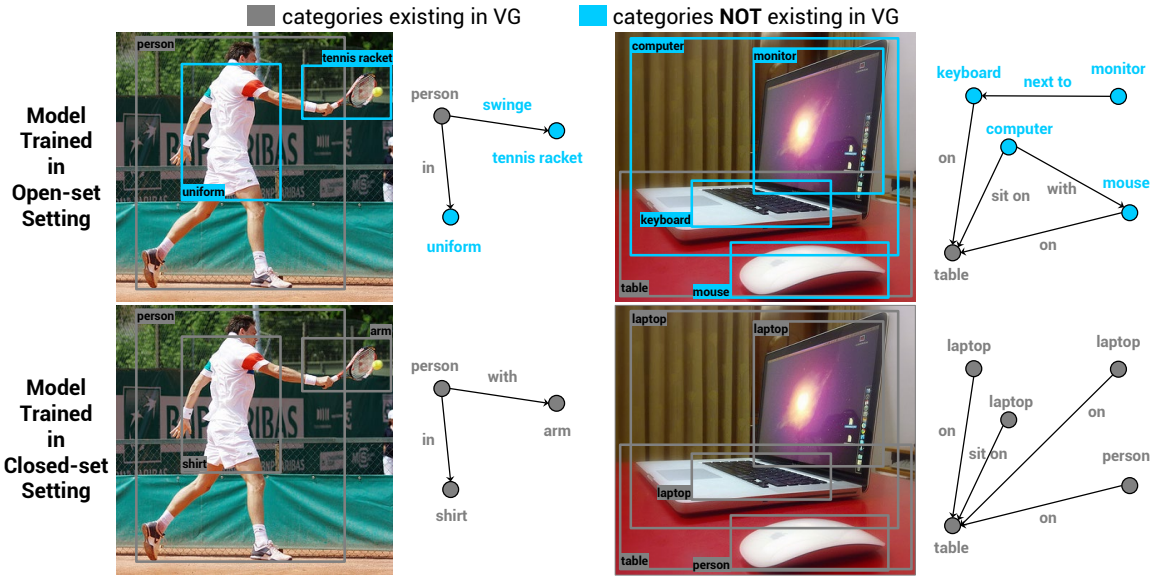


Figure 4.5: Qualitative results of our models (trained in open-set and closed-set settings) on VG test set for SGG.

4.4 Conclusion

In this chapter, we proposed one of the first methods of learning to generate scene graphs from image-text pairs. Our key idea is to use off-the-shelf object detectors, so as to match detected object tags to parsed tokens from captions, thus creating "pseudo" labels for training. Further, we designed a Transformer-based model and demonstrated strong results across different levels of supervision. Our model learned from only image-text pairs, outperformed a state-of-the-art weakly supervised model trained by human-annotated unlocalized scene graphs. More importantly, we presented the first result for open-set scene graph generation. We hope our work points to exciting avenues of learning structured visual representation from natural language.

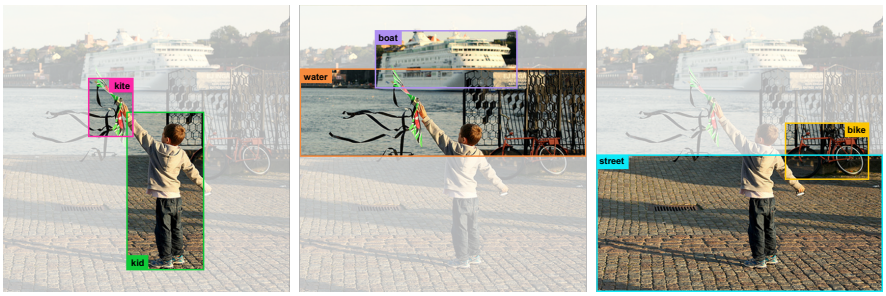
5 LEARNING SCENE COMPONENT

Chapter 4 demonstrated my work of learning object relationships from image-text pairs, thereby generating scene graphs for input images. A natural question is: *how can symbolic representation like scene graph help other vision tasks?* In this chapter, we explore the benefits of leveraging scene graph structure for image captioning.

It is an old saying that "A picture is worth a thousand words". Complex and sometimes multiple ideas can be conveyed by a single image. Consider the example in Fig. 5.1. The image can be described by "A boy is flying a kite" when pointing to the boy and the kite, or depicted as "A ship is sailing on the river" when attending to the boat and the river. Instead, when presented with regions of the bike and the street, the description can be "A bike parked on the street". Humans demonstrate remarkable ability to summarize multiple ideas associated with different scene components in the same image. More interestingly, we can easily explain our descriptions by linking sentence tokens back to image regions.

Despite recent progress in image captioning, most of current approaches are optimized for caption quality. These methods tend to produce generic sentences that are minorly reworded from those in the training set, and to "look" at regions that are irrelevant to the output sentence Das et al. (2017); Rohrbach et al. (2018). Several recent efforts seek to address these issues, leading to models designed for individual tasks including diverse Wang et al. (2017); Deshpande et al. (2019), grounded Selvaraju et al. (2019); Zhou et al. (2019) and controllable captioning Lu et al. (2018); Cornia et al. (2019). However, no previous method exists that can address diversity, grounding, and controllability at the same time in a single model — abilities seemingly effortless for we humans.

In this chapter, we seek to bridge the gap between captioning models and humans, and enable multiple capabilities of image captioning within single model. To this end, we propose to revisit an image representation that can better link image regions to sentence descriptions—scene graph. The key idea is that such a graph can be decomposed into a set of sub-graphs, with each sub-graph as a candidate scene component that might be described by a unique sentence. Our goal is thus to design a model that can identify meaningful sub-graphs and decode their corresponding descriptions. A major advantage of this design is that diversity and controllability are naturally enabled by selecting multiple distinct sub-graphs to decode and by



(a) A young boy is flying a kite. (b) A cruise ship is sailing on the river. (c) A bike is parked on the street.

Figure 5.1: An example image with multiple scene components with each described by a distinct caption. *How can we design a model that can learn to identify and describe different components of an input image?*

specifying a set of sub-graphs for sentence generation.

Specifically, our method takes a scene graph extracted from an image as input. This graph consists of nodes as objects (nouns) and edges as the relations between pairs of objects (predicates). Each node or edge comes with its text and visual features. Our method first constructs a set of overlapping sub-graphs from the full graph. We develop a graph neural network that learns to select meaningful sub-graphs best described by one of the human annotated sentences. Each of the selected sub-graphs is further decoded into its corresponding sentence. This decoding process incorporates an attention mechanism on the sub-graph nodes when generating each token. Our model thus supports backtracking of generated sentence tokens into scene graph nodes and its image regions. Consequently, our method provides the *first* comprehensive model for generating accurate, diverse, and controllable captions that are grounded into image regions.

This chapter is organized as follows. In Section 5.1, I summarize our contributions. In Section 5.2 and Section 5.3, I present our method and key experiments, respectively. Finally, I conclude the chapter in Section 5.4.

5.1 Contributions

The contributions of this chapter are summarized into three folds:

- We propose the first comprehensive image captioning method that enables accurate, diverse, grounded and controllable captioning at the same time.

- The key innovation of our method is learning to identify important scene components by selecting sub-graphs from an input scene graph, and to decode each component into a sentence description.
- Our model establishes new state-of-the-art results in diverse captioning, grounded captioning and controllable captioning, and compares favourably to the latest methods optimized for caption quality.

This work was a collaboration with Liwei Wang, Jianshu Chen, Dong Yu, and Yin Li. The work has been published by ECCV 2020 with title as "Comprehensive Image Captioning via Scene Graph Decomposition" Zhong et al. (2020).

5.2 Method

Given an input image I, we assume an image scene graph $G = (V, E)$ can be extracted from I, where V represents the set of nodes corresponding to the detected objects (nouns) in I, and E represents the set of edges corresponding to the relationships between pairs of objects (predicates). Our goal is to generate a set of sentences $C = \{C_j\}$ to describe different components of I using the scene graph G. To this end, we propose to make use of the sub-graphs $\{G_i^s = (V_i^s, E_i^s)\}$ from G, where $V_i^s \subseteq V$ and $E_i^s \subseteq E$. Our method seeks to model the joint probability $P(S_{ij} = (G, G_i^s, C_j)|I)$, where $P(S_{ij}|I) = 1$ indicates that the sub-graph G_i^s can be used to decode the sentence C_j. Otherwise, $P(S_{ij}|I) = 0$. We further assume that $P(S_{ij}|I)$ can be decomposed into three parts, given by

$$P(S_{ij}|I) = P(G|I)P(G_i^s|G, I)P(C_j|G_i^s, G, I). \tag{5.1}$$

Intuitively, $P(G|I)$ extracts scene graph G from an input image I. $P(G_i^s|G, I)$ decomposes the full graph G into a diverse set of sub-graphs $\{G_i^s\}$ and selects important sub-graphs for sentence generation. Finally, $P(C_j|G_i^s, G, I)$ decodes a selected sub-graph G_i^s into its corresponding sentence C_j, and also associates the tokens in C_j to the nodes V_i^s of the sub-graph G_i^s (the image regions in I). Fig. 5.2 illustrates the details of our method.

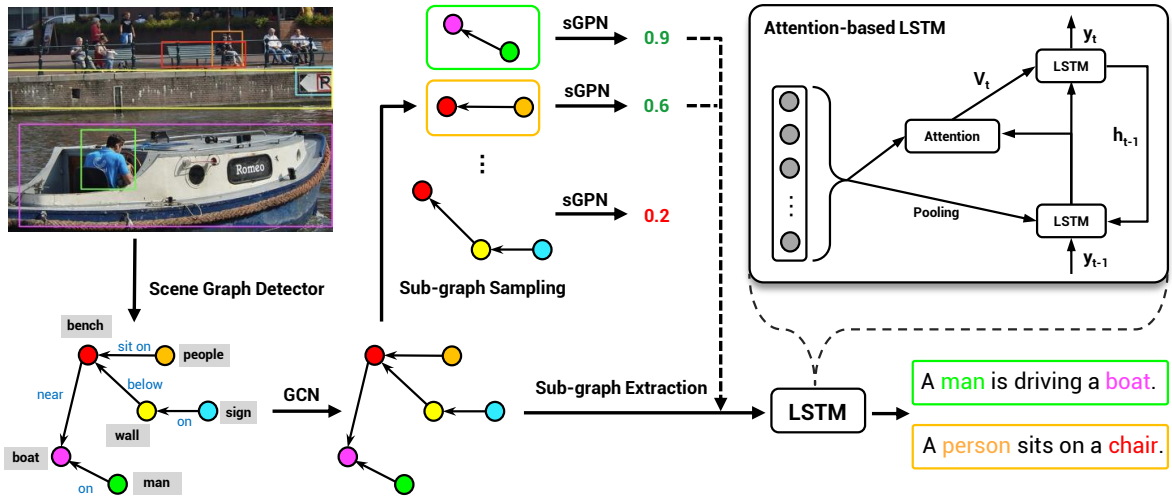


Figure 5.2: Overview of our method. Our method takes a scene graph extracted from an input image, and decomposes the graph into a set of sub-graphs. We design a sub-graph proposal network (sGPN) that learns to identify meaningful sub-graphs, which are further decoded by an attention-based LSTM for generating sentences and grounding sentence tokens into sub-graph nodes (image regions). By leveraging sub-graphs, our method for the first time unifies accurate, diverse, grounded, and controllable image captioning in a single model.

Scene Graph Detection and Decomposition

Our method first extracts scene graph G from image I ($P(G|I)$) using MotifNet Zellers et al. (2018). MotifNet builds LSTMs on top of object detector outputs Ren et al. (2015b) and produces a scene graph $G = (V, E)$ with nodes V for common objects (nouns) and edges E for relationship between pairs of objects (predicates), such as "holding","behind" or "made of". Note that G is a directed graph, i.e., an edge must start from a subject noun or end at an object noun. Therefore, the graph G is defined by a collection of subject-predicate-object triplets, e.g., kid playing ball.

We further samples sub-graphs $\{G_i^s\}$ from the scene graph G by using neighbor sampling Klusowski and Wu (2018). Specifically, we randomly select a set of seed nodes $\{S_i\}$ on the graph. The immediate neighbors of the seed nodes with the edges in-between define a sampled sub-graph. Formally, the sets of sub-graph nodes and edges are $V_i^s = S_i \cup \{N(v)|v \in S_i\}$ and $E_i^s = \{(v, u)|v \in S_i, u \in N(v)\}$ respectively, where $N(v)$ denotes the immediate neighbors of node v. Identical sub-graphs are removed to obtain the final set of sub-graphs $\{G_i^s = (V_i^s, E_i^s)\}$, which covers potential scene components in the input image I.

Sub-graph Proposal Network

Our next step is to identify meaningful sub-graphs that are likely to capture major scene components in the image ($P(G_i^s|G, I)$). Specifically, our model first combines visual and text features on the scene graph G, followed by an integration of contextual information within G using a graph convolutional network, and finally a score function learned to rank sub-graphs G_i^s.

Scene Graph Representation. Given a directed scene graph $G = (V, E)$, we augment its nodes and edges with visual and text features. For a node $v \in V$, we use both its visual feature extracted from image regions and the word embedding of its noun label. We denote the visual features as $\mathbf{x}_v^v \in \mathbb{R}^{d_v}$ and text features as $\mathbf{x}_v^e \in \mathbb{R}^{d_e}$. For an edge $e \in E$, we only use the embedding of its predicate label denoted as $\mathbf{x}_e^e \in \mathbb{R}^{d_e}$. Subscripts are used to distinguish node (v) and edge (e) features and superscripts to denote the feature type, i.e., visual features or text embedding. Visual and text features are further fused by projecting them into a common sub-space. This is done separately for node and edge features by

$$\mathbf{x}_v^f = \mathrm{ReLU}(\mathbf{W}_f^1 \mathbf{x}_v^v + \mathbf{W}_f^2 \mathbf{x}_v^e), \quad \mathbf{x}_e^f = \mathbf{W}_f^3 \mathbf{x}_e^e, \tag{5.2}$$

where $\mathbf{W}_f^1 \in \mathbb{R}^{d_f \times d_v}$, $\mathbf{W}_f^2 \in \mathbb{R}^{d_f \times d_e}$ and $\mathbf{W}_f^3 \in \mathbb{R}^{d_f \times d_e}$ are learned projections.

Graph Convolutional Network (GCN). After feature fusion and projection, we further model the context between objects and their relationships using a GCN. The GCN aggregates information from the neighborhood within the graph and updates node and edge features. With an proper ordering of the nodes and edges, we denote the feature matrix for nodes and edges as $\mathbf{X}_v^f = [\mathbf{x}_v^f] \in \mathbb{R}^{d_f \times |V|}$ and $\mathbf{X}_e^f = [\mathbf{x}_e^f] \in \mathbb{R}^{d_f \times |E|}$, respectively. The update rule of a single graph convolution is thus given by

$$\begin{aligned} \hat{\mathbf{X}}_v^f &= \mathbf{X}_v^f + \mathrm{ReLU}(\mathbf{W}_{ps} \mathbf{X}_e^f \mathbf{A}_{ps}) + \mathrm{ReLU}(\mathbf{W}_{po} \mathbf{X}_e^f \mathbf{A}_{po}), \\ \hat{\mathbf{X}}_e^f &= \mathbf{X}_e^f + \mathrm{ReLU}(\mathbf{W}_{sp} \mathbf{X}_v^f \mathbf{A}_{sp}) + \mathrm{ReLU}(\mathbf{W}_{op} \mathbf{X}_v^f \mathbf{A}_{op}), \end{aligned} \tag{5.3}$$

where $\mathbf{W}_{ps}, \mathbf{W}_{po}, \mathbf{W}_{sp}, \mathbf{W}_{op} \in \mathbb{R}^{d_f \times d_f}$ are learnable parameters that link subject or object features (nouns) with predicate features. For example, $\mathbf{W}_{ps}$ connects between predicate features and subject features. $\mathbf{A}_{ps}, \mathbf{A}_{po} \in \mathbb{R}^{|E| \times |V|}$ are the normalized adjacency matrix (defined by G) between predicates and subjects, and between predicates and objects, respectively. For instance, a non-zero element in $\mathbf{A}_{ps}$ suggests a link between a predicate and a subject on the scene graph G. Similarly, $\mathbf{A}_{sp}, \mathbf{A}_{op} \in$

$\mathbb{R}^{|V|\times|E|}$ are the normalized adjacency matrix between subjects and predicates, and between objects and predicates.

Our GCN stacks several graph convolutions, and produces an output scene graph with updated node and edge features. We only keep the final node features ($\mathbf{X}_v^u = [\mathbf{x}_v^u], v \in V$) for subsequent sub-graph ranking, as the predicate information has been integrated using GCN.

Sub-graph Score Function. With the updated scene graph and the set of sampled sub-graphs, our model learns a score function to select meaningful sub-graphs for generating sentence descriptions. For each sub-graph, we index its node features as $\mathbf{X}_i^s = [\mathbf{x}_v^u], v \in V_i^s$ and construct a score function

$$s_i = \sigma(f(\Phi(\mathbf{X}_i^s))), \tag{5.4}$$

where $\Phi(\cdot)$ is a sub-graph readout function Xu et al. (2019) that concatenates the max-pooled and mean-pooled node features on the sub-graph. $f(\cdot)$ is a score function realized by a two-layer multilayer perceptron (MLP). And $\sigma(\cdot)$ is a sigmoid function that normalizes the output score into the range of $[0, 1]$.

Learning the Score Function. The key challenge of learning the score function f is the training labels. Our goal is to rank the sampled sub-graphs and select the best ones to generate captions. Thus, we propose to use ground-truth captions provided by human annotators to guide the learning. A sub-graph with most of the nodes matched to one of the ground-truth sentences should be selected. To this end, we recast the learning of the score function f as training a binary classifier to distinguish between "good" (positive) and "bad" (negative) sub-graphs. Importantly, we design a matching score between a ground-truth sentence and a sampled sub-graph to generate the target binary labels, so as to train our binary classifier.

Specifically, given a sentence C_j and a scene graph G, we extract a reference sub-graph on G by finding the nodes on the graph G that also appears in the sentence C_j and including their immediate neighbor nodes. This is done by extracting nouns from the sentence C_j using a part-of-speech tag parser Bird and Loper (2004), and matching the nouns to the nodes on G using LCH score Leacock et al. (1998) derived from WordNet Miller (1995). This matching process is given by $M(C_j, G)$. We further compute the node Intersection over Union (IoU) score between the reference sub-

graph $M(C_j, G)$ and each of the sampled sub-graph G_i^s by

$$\text{IoU}(G_i^s, C_j) = \frac{|G_i^s \cap M(C_j, G)|}{|G_i^s \cup M(C_j, G)|}, \tag{5.5}$$

where $\cap$ and $\cup$ are the intersection and union operation over sets of sub-graph nodes, respectively. The node IoU provides a matching score between the reference sentence C_j and the sub-graph G_i^s and is used to determine our training labels. We only consider a sub-graph as positive for training if its IoU with any of the target sentences is higher than a pre-defined threshold (0.75).

Training Strategy. A major issue in training is that we have many negative sub-graphs and only a few positive ones. To address this issue, a mini-batch of sub-graphs is randomly sampled to train our sGPN, where positive to negative ratio is kept as 1:1. If a ground-truth sentence does not match any positive sub-graph, we use the reference sub-graph from $M(C_j, G)$ as its positive sub-graph.

Decoding Sentences from Sub-graphs

Our final step is to generate a target sentence using features from any selected single sub-graph $(P(C_j|G_i^s, G, I))$. We modify the attention-based LSTM Anderson et al. (2018) for sub-graph decoding, as shown in Fig. 3.2 (top right). Specifically, the model couples an attention LSTM and a language LSTM. The attention LSTM assigns each sub-graph node an importance score, further used by the language LSTM to generate the tokens. Specifically, at each time step t, the attention LSTM is given by $\mathbf{h}_t^A = \text{LSTM}_{Att}([\mathbf{h}_{t-1}^L, \mathbf{e}_t, \mathbf{x}_i^s])$, where $\mathbf{h}_{t-1}^L$ is the hidden state of the language LSTM at time $t-1$. e_t is the word embedding of the input token at time t and $\mathbf{x}_i^s$ is the sub-graph feature. Instead of averaging all region features as Anderson et al. (2018); Yao et al. (2018), our model uses the input sub-graph feature, given by $\mathbf{x}_i^s = g(\Phi(\mathbf{X}_i^s))$, where $g(\cdot)$ is a two-layer MLP, $\Phi(\cdot)$ is the same graph readout unit in Eq. 5.4.

Based on hidden states $\mathbf{h}_t^A$ and the node features $\mathbf{X}_i^s = [\mathbf{x}_v^u]$ in the sub-graph, an attention weight $a_{v,t}$ at time t for node v is computed by $a_{v,t} = \mathbf{w}_a^T \tanh(\mathbf{W}_v \mathbf{x}_v^u + \mathbf{W}_h \mathbf{h}_t^A)$ with learnable weights $\mathbf{W}_v$, $\mathbf{W}_h$ and $\mathbf{w}_a$. A softmax function is further used to normalize $\mathbf{a}_t$ into $\mathbf{ff}_t$ defined on all sub-graph nodes at time t. We use $\mathbf{ff}_t$ to backtrack image regions associated with a decoded token for caption grounding. Finally, the hidden state of the attention LSTM $\mathbf{h}_t^A$ and the attention re-weighted sub-graph feature $\mathbf{V}_t = \sum_v \alpha_{v,t} \mathbf{x}_v^u$ are used as the input of the language LSTM—a standard

LSTM that decodes the next word.

Training and Inference

We summarize the training and inference schemes of our model.

Loss Functions. Our sub-graph captioning model has three parts: $P(G|I)$, $P(G_i^s|G, I)$, $P(C_j|G_i^s, G, I)$, where the scene graph generation $(P(G|I))$ is trained independently on Visual Genome Krishna et al. (2017). For training, we combine two loss functions for $P(G_i^s|G, I)$ and $P(C_j|G_i^s, G, I)$. Concretely, we use a binary cross-entropy loss for the sub-graph proposal network $(P(G_i^s|G, I))$, and a multi-way cross-entropy loss for the attention-based LSTM model to decode the sentences $(P(C_j|G_i^s, G, I))$. The coefficient between the two losses is set to 1.

Inference. During inference, our model extracts the scene graph, samples sub-graphs and evaluates their sGPN scores. *Greedy Non-Maximal Suppression* (*NMS*) is further used to filter out sub-graphs that largely overlap with others, and to keep sub-graphs with high sGPN scores. The overlapping between two sub-graphs is defined by the IoU of their nodes. We find that using NMS during testing helps to remove redundant captions and to promote diversity.

After NMS, top-ranked sub-graphs are decoded using an attention-based LSTM. As shown in Luo and Shakhnarovich (2020), an *optional top-K sampling* Fan et al. (2018); Radford et al. (2019) can be applied during the decoding to further improve caption diversity. We disable top-K sampling for our experiments unless otherwise noticed. The final output is thus a set of sentences with each from a single sub-graph. By choosing which sub-graphs to decode, our model can control caption contents. Finally, we use attention weights in the LSTM to ground decoded tokens to sub-graph nodes (image regions).

5.3 Key Experiments

In this section, I describe the implementation details and present results. Our model is evaluated across several captioning tasks, including accurate and diverse captioning, grounded captioning, and controllable captioning.

Implementation Details. We used Faster R-CNN Ren et al. (2015b) with ResNet-101 He et al. (2016) from Anderson et al. (2018) as our object detector. Based on detection results, Motif-Net Zellers et al. (2018) was trained on Visual Genome Kr-

ishna et al. (2017) with 1600/20 object/predicate classes. For each image, we applied the detector and kept 36 objects and 64 triplets in scene graph. We sampled 1000 sub-graphs per image and removed duplicate ones, leading to an average of 255/274 sub-graphs per image for MS-COCO Chen et al. (2015)/Flickr30K Plummer et al. (2015). We used 2048D visual features for image regions and 300D GloVe Pennington et al. (2014) embeddings for node and edge labels. These features were projected into 1024D, followed by a GCN with depth of 2 for feature transform and an attention LSTM (similar to Anderson et al. (2018)) for sentence decoding. For training, we used Adam Kingma and Ba (2015) with initial learning rate of 0.0005 and a mini-batch of 64 images and 256 sub-graphs. Beam search was used in decoding with beam size 2, unless otherwise noted.

Accurate and Diverse Image Captioning

Dataset and Metric. We follow the evaluation protocol from Vijayakumar et al. (2018); Wang et al. (2017); Deshpande et al. (2019); Aneja et al. (2019) and report both accuracy and diversity results using the M-RNN split Mao et al. (2015) of MS-COCO Caption dataset Chen et al. (2015). Specifically, this split has 118,287/4,000/1,000 images for train/val/test set, with 5 human labeled captions per image. We train the model on the train set and report the results on the *val* set. For accuracy, we report top-1 accuracy out of the top 20/100 output captions, using BLEU Papineni et al. (2002), CIDEr Vedantam et al. (2015), ROUGE-L Lin (2004), METEOR Banerjee and Lavie (2005) and SPICE Anderson et al. (2016). For diversity, we evaluate the percentage of distinct captions from 20/100 sampled output captions and report the scores for novel sentences, mutual overlap (mBLEU-4), and 1/2-gram diversity of the best 5 sampled captions using a ranking function.

Baselines. We consider several latest methods designed for diverse and accurate captioning as our baselines, including Div-BS Vijayakumar et al. (2018), AG-CVAE Wang et al. (2017), POS Deshpande et al. (2019), POS+Joint Deshpande et al. (2019) and Seq-CVAE Aneja et al. (2019). We compare our results of Sub-GC to these baselines in Table 5.1. In addition, we include the results of our model with top-K sampling (Sub-GC-S), as well as human performance for references of diversity.

Diversity Results. For the majority of the diversity metrics, our model Sub-GC outperforms previous methods (+8% for novel sentences and +29%/20% for 1/2-gram with 20 samples), except the most recent Seq-CVAE. Upon a close inspection of

Method	#	Diversity					Top-1 Accuracy							
		Distinct Caption (↑)	#novel (Best 5) (↑)	mBLEU-4 (Best 5) (↓)	1-gram (Best 5) (↑)	2-gram (Best 5) (↑)	B1	B2	B3	B4	C	R	M	S
Div-BS Vijayakumar et al. (2018)	20	100%	3106	81.3	0.20	0.26	72.9	56.2	42.4	32.0	103.2	53.6	25.5	18.4
AG-CVAE Wang et al. (2017)		69.8%	3189	66.6	0.24	0.34	71.6	54.4	40.2	29.9	96.3	51.8	23.7	17.3
POS Deshpande et al. (2019)		96.3%	3394	63.9	0.24	0.35	74.4	57.0	41.9	30.6	101.4	53.1	25.2	18.8
POS+Joint Deshpande et al. (2019)		77.9%	3409	66.2	0.23	0.33	73.7	56.3	41.5	30.5	102.0	53.1	25.1	18.5
Sub-GC		71.1%	3679	67.2	0.31	0.42	**77.2**	**60.9**	**46.2**	**34.6**	**114.4**	**56.1**	**26.9**	**20.0**
Seq-CVAE Aneja et al. (2019)	20	94.0%	**4266**	52.0	0.25	0.54	73.1	55.4	40.2	28.9	100.0	52.1	24.5	17.5
Sub-GC-S		**96.2%**	4153	**36.4**	**0.39**	**0.57**	75.2	57.6	42.7	31.4	107.3	54.1	26.1	19.3
Div-BS Vijayakumar et al. (2018)	100	100%	3421	82.4	0.20	0.25	73.4	56.9	43.0	32.5	103.4	53.8	25.5	18.7
AG-CVAE Wang et al. (2017)		47.4%	3069	70.6	0.23	0.32	73.2	55.9	41.7	31.1	100.1	52.8	24.5	17.9
POS Deshpande et al. (2019)		91.5%	3446	67.3	0.23	0.33	73.7	56.7	42.1	31.1	103.6	53.0	25.3	18.8
POS+Joint Deshpande et al. (2019)		58.1%	3427	70.3	0.22	0.31	73.9	56.9	42.5	31.6	104.5	53.2	25.5	18.8
Sub-GC		65.8%	3647	69.0	0.31	0.41	**77.2**	**60.9**	**46.2**	**34.6**	**114.4**	**56.1**	**26.9**	**20.0**
Seq-CVAE Aneja et al. (2019)	100	84.2%	**4215**	64.0	0.33	0.48	74.3	56.8	41.9	30.8	104.1	53.1	24.8	17.8
Sub-GC-S		**94.6%**	4128	**37.3**	**0.39**	**0.57**	75.2	57.6	42.7	31.4	107.3	54.1	26.1	19.3
Human	5	99.8%	-	51.0	0.34	0.48	-	-	-	-	-	-	-	-

Table 5.1: Diversity and top-1 accuracy results on COCO Caption dataset (M-RNN split Mao et al. (2015)). Best-5 refers to the top-5 sentences selected by a ranking function. Note that Sub-GC and Sub-GC-S have same top-1 accuracy in terms of sample-20 and sample-100, since we have a sGPN score per sub-graph and global sorting is applied over all sampled sub-graphs. Our models outperform previous methods on both top-1 accuracy and diversity for the majority of the metrics.

Seq-CVAE model, we hypothesis that Seq-CVAE benefits from sampling tokens at each time step. It is thus meaningful to compare our model using top-K sampling (Sub-GC-S) with Seq-CVAE. Sub-GC-S outperforms Seq-CVAE in most metrics (+18%/19% for 1/2-gram with 100 samples) and remains comparable for the metric of novel sentences (within 3% difference).

Accuracy Results. We notice that the results of our sub-graph captioning models remain the same with increased number of samples. This is because our outputs follow a fixed rank from sGPN scores. Our Sub-GC outperforms all previous methods by a significant margin. Sub-GC achieves +2.6/2.1 in B4 and +11.2/9.9 in CIDEr when using 20/100 samples in comparison to previous best results. Moreover, while achieving best diversity scores, our model with top-K sampling (Sub-GC-S) also outperforms previous methods in most accuracy metrics (+0.8/0.9 in B1 and +4.1/2.8 in CIDEr when using 20/100 samples) despite its decreased accuracy from Sub-GC.

Comparison to Accuracy Optimized Captioning models. We conduct further experiments to compare the top ranked sentence from ou Sub-GC against the results of latest captioning models optimized for accuracy, including Up-Down Anderson et al. (2018), GCN-LSTM Yao et al. (2018) and SGAE Yang et al. (2019). All these previous models can only generate a single sentence, while our method (Sub-GC) can generate a set of diverse captions. As a reference, we consider a variant of our model (Full-GC) that uses a full scene graph instead of sub-graphs to decode sentences. Moreover, we

Method	B1	B4	C	R	M	S
Up-Down Anderson et al. (2018)	77.2	36.2	113.5	56.4	27.0	20.3
GCN-LSTM Yao et al. (2018)	77.3	36.8	116.3	57.0	**27.9**	**20.9**
SGAE Yang et al. (2019)	**77.6**	**36.9**	**116.7**	**57.2**	27.7	**20.9**
Full-GC	76.7	**36.9**	114.8	56.8	**27.9**	20.8
Sub-GC	76.8	36.2	115.3	56.6	27.7	20.7
Sub-GC-oracle	90.7	59.3	166.7	71.5	40.1	30.1

Table 5.2: Comparison to accuracy optimized models on COCO caption dataset using Karpathy split Karpathy and Fei-Fei (2015). Our Sub-GC compares favorably to the latest methods that were designed to only output a single high-quality caption.

include an upper bound of our model (Sub-GC-oracle) by assuming that we have an oracle ranking function, i.e., always selecting the maximum scored sentence for each metric. All results are reported on Karpathy split Karpathy and Fei-Fei (2015) without using reinforcement learning for score optimization Rennie et al. (2017).

Our results are shown in Table 5.2. Our Sub-GC achieves comparable results (within 1-2 points in B4/CIDEr) to latest methods (Up-Down, GCN-LSTM and SGAE). We find that the results of our sub-graph captioning model is slightly worse than those models using the full scene graph, e.g., Full-GC, GCN-LSTM and SGAE. We argue that this minor performance gap does not diminish our contribution, as our model offers new capacity for generating diverse, controllable and grounded captions. Notably, our best case (Sub-GC-oracle) outperforms all other methods for all metrics by a very large margin (+22.4 in B4 and +50.0 in CIDEr). These results suggest that at least one high-quality caption exists among the sentences decoded from the sub-graphs. It is thus possible to generate highly accurate captions if there is a way to select this "good" sub-graph.

Grounded Image Captioning

Moreover, we evaluate our model for grounded captioning. We describe the dataset and metric, introduce our setup and baselines, and discuss our results.

Dataset and Metric. We use Flickr30k Entities Plummer et al. (2015) for grounded captioning. Flickr30k Entities has 31K images, with 5 captions for each image. The dataset also includes 275k annotated bounding boxes associated with the phrases in corresponding captions. We use the data split from Karpathy and Fei-Fei (2015). To evaluate the grounding performance, we follow the protocol in GVD Zhou et al. (2019). We report both $F1_{all}$ and $F1_{loc}$. $F1_{all}$ considers a region prediction as correct

Method	Grounding Evaluation		Caption Evaluation				
	F1 all	F1 loc	B1	B4	C	M	S
GVD Zhou et al. (2019)	3.88	11.70	69.2	26.9	60.1	22.1	16.1
Up-Down Anderson et al. (2018)	4.14	12.30	69.4	27.3	56.6	21.7	16.0
Cyclical Ma et al. (2019)	4.98	13.53	69.9	27.4	61.4	22.3	16.6
Full-GC	4.90	13.08	69.8	**29.1**	**63.5**	**22.7**	**17.0**
Sub-GC	**5.98**	**16.53**	**70.7**	28.5	61.9	22.3	16.4
GVD (Sup.) Zhou et al. (2019)	7.55	22.20	69.9	27.3	62.3	22.5	16.5

Table 5.3: Grounded captioning results on Flickr30K Entities Plummer et al. (2015). Our method (Sub-GC) outperforms previous weakly supervised methods.

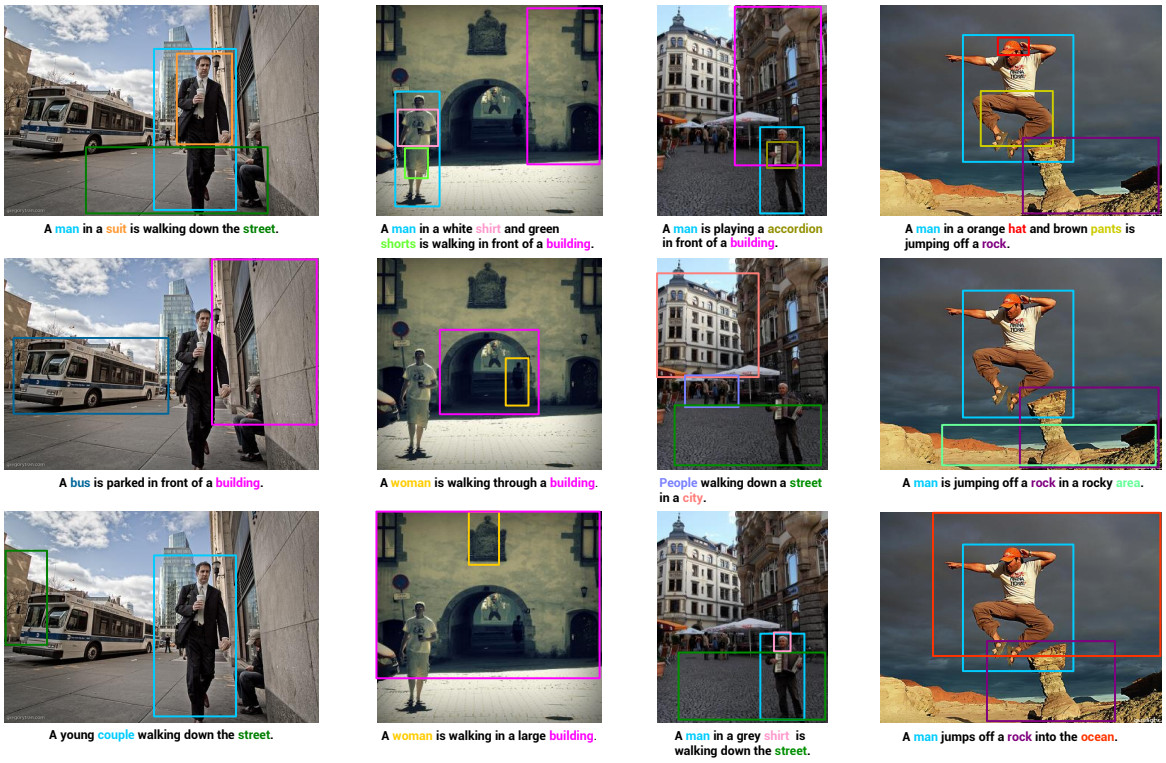


Figure 5.3: Sample results of our Sub-GC on Flickr30k Entities test set. Each column shows three captions with their region groundings decoded from different sub-graphs for an input image. The first two rows are successful cases and the last row is the failure case. These sentences can describe different parts of the images. Each generated noun and its grounding bounding box are highlighted in the same color.

if the object word is correctly predicted and the box is correctly localized. On the other hand $F1_{loc}$ only accounts for localization quality. Moreover, we report the standard BLEU Papineni et al. (2002), CIDEr Vedantam et al. (2015), METEOR Banerjee and Lavie (2005) and SPICE Anderson et al. (2016) scores for caption quality.

Experiment Setup and Baselines. For this experiment, we only evaluate the top-

ranked sentence and its grounding from our model. We select the node on the sub-graph with maximum attention weight when decoding a noun word, and use its bounding box as the grounded region. Our results are compared to a strong set of baselines designed for weakly supervised grounded captioning, including weakly supervised GVD Zhou et al. (2019), Up-Down Anderson et al. (2018) and a concurrent work Cyclical Ma et al. (2019). We also include reference results from fully supervised GVD Zhou et al. (2019) that requires ground-truth matching pairs for training, and our Full-GC that decode a sentence from a full graph.

Results. Our results are presented in Table 5.3. Among all weakly supervised methods, our model achieves the best F1 scores for caption grounding. Specifically, our sub-graph captioning model (Sub-GC) outperforms previous best results by +1.0 for $F1_{all}$ and +3.0 for $F1_{loc}$, leading to a relative improvement of 20% and 22% for $F1_{all}$ and $F1_{loc}$, respectively. Our results also have the highest captioning quality (+1.1 in B4 and +0.5 in CIDEr). We conjecture that constraining the attention to the nodes of a sub-graph helps to improve the grounding. Fig. 5.3 shows sample results of grounded captions. Not surprisingly, the supervised GVD outperforms our Sub-GC. Supervised GVD can be considered as an upper bound for all other methods, as it uses grounding annotations for training. Comparing to our Full-GC, our Sub-GC is worse on captioning quality (-0.6 in B4 and -1.6 in CIDEr) yet has significant better performance for grounding (+1.1 in $F1_{all}$ and +3.5 in $F1_{loc}$).

Controllable Image Captioning

Finally, we report results on controllable image captioning. Again, we describe our experiments and present the results.

Dataset and Metric. Same as grounding, we consider Flickr30k Entities Plummer et al. (2015) for controllable image captioning and use the data split Karpathy and Fei-Fei (2015). We follow evaluation protocol developed in Cornia et al. (2019). Specifically, the protocol assumes that an image and a set of regions are given as input, and evaluates a decoded sentence against one or more target ground-truth sentences. These ground-truth sentences are selected from captions by matching the sentences tokens to object regions in the image. Standard captioning metrics are considered (BLEU Papineni et al. (2002), CIDEr Vedantam et al. (2015), ROUGE-L Lin (2004), METEOR Banerjee and Lavie (2005) and SPICE Anderson et al. (2016)), yet the ground-truth is different from conventional image captioning. Further, IoU of nouns

Method	B1	B4	C	R	M	S	IoU
NBT Lu et al. (2018) (Sup.)	-	8.6	53.8	31.9	13.5	17.8	49.9
SCT Cornia et al. (2019) (Sup.)	33.1	9.9	67.3	35.3	14.9	22.2	52.7
Sub-GC	33.6	9.3	57.8	32.5	14.2	18.8	50.6
Sub-GC (Sup.)	**36.2**	**11.2**	**73.7**	**35.5**	**15.9**	**22.2**	**54.1**

Table 5.4: Controllable captioning results on Flickr30K Entities Plummer et al. (2015). With weak supervision, our Sub-GC compares favorably to previous methods. With strong supervision, our Sub-GC (Sup.) achieves the best results.

between the predicted and the target sentence is reported as Cornia et al. (2019).

Experiment Setup and Baselines. We consider (1) our Sub-GC trained with only image-sentence pairs; and (2) a supervised Sub-GC trained with ground-truth pairs of region sets and sentences as Cornia et al. (2019). Both models follow the same inference scheme, where input controlled set of regions are converted into best matching sub-graphs for sentence decoding. However, supervised Sub-GC uses these matching during training. We compare our results to recent methods developed for controllable captioning, including NBT Lu et al. (2018) and SCT Cornia et al. (2019). NBT and SCT are trained with matching pairs of region sets and sentences same as our supervised Sub-GC. Results are reported without using reinforcement learning.

Results. The results are shown in Table 5.4. Our models demonstrate strong controllability of the output sentences. Specifically, our supervised Sub-GC outperforms previous supervised methods (NBT and SCT) by a significant margin. Comparing to previous best SCT, our results are +1.3 in B4, +6.4 in CIDEr and +1.4 in IoU. Interestingly, our vanilla model has comparable performance to previous methods, even if it is trained with only image sentence pairs. These results provide further supports to our design of using sub-graphs for image captioning.

5.4 Conclusion

In this chapter, we proposed a novel image captioning model by exploring sub-graphs of image scene graph. Our key idea is to select important sub-graphs and only decode a single target sentence from a selected sub-graph. We demonstrated that our model can generate accurate, diverse, grounded and controllable captions. Our method thus offers the first comprehensive model for image captioning. Moreover, our results established new state-of-the-art in diverse captioning, grounded captioning and controllable captioning, and compared favourably to latest method for caption quality.

We hope our work can provide insights into the design of explainable and controllable models for vision and language tasks.

6 LEARNING ACTION PROCEDURE

In chapter 3, chapter 4, and chapter 5, I introduced my work on learning visual knowledge for static images by using image-text pairs as training data. However, the visual world by its nature is dynamic, full of actions and can be recorded in videos. In this chapter, I extend the key idea of language-guided visual learning to video understanding. More specifically, we propose a method to learn action procedures from video-narration pairs for procedural activity understanding.

Many of our daily activities (*e.g.*, cooking or crafting) are highly structured, comprising a set of action steps conducted in a certain ordering. Yet how these activities are performed varies among individuals. Consider the example of making scrambled eggs as shown in Fig. 6.1. While most people tend to whisk eggs in a bowl, melt butter in a pan, and cook eggs under medium heat, expert chefs have recommended to crack eggs into the pan, add butter, and stir them under high heat. Imagine a vision model that can account for the individual variations and reason about the temporal ordering of action steps in a video, so as to infer prior missing steps, recognize the current step, and forecast a future step. Such a model will be immensely useful for a wide range of applications including augmented reality, virtual personal assistants, and human-robot interaction.

Understanding complex procedural activities has been a long-standing challenge in the vision community Pei et al. (2011); Gupta et al. (2009); Ryoo and Aggarwal (2006); Ivanov and Bobick (2000); Brand et al. (1997); Nevatia et al. (2003). While many prior approaches learn from annotated videos following a fully supervised setting Kuehne et al. (2014); Zhou et al. (2018b); Elhamifar and Naing (2019), this paradigm is difficult to scale to a plethora of activities and their variants among individuals. A promising solution is offered by the exciting advances in vision-and-language pre-training, where models learn from visual data (images or videos) and their paired text data (captions or narrations) Radford et al. (2021); Li et al. (2021); Sun et al. (2019); Zhu and Yang (2020) in order to recognize a variety of concepts. This idea has recently been explored to analyze instructional videos Miech et al. (2020); Lin et al. (2022), yet existing methods are limited to recognizing single action steps in procedural activities.

In this chapter, we present a first step towards modeling the temporal ordering of action steps in procedural activities by learning from instructional videos and

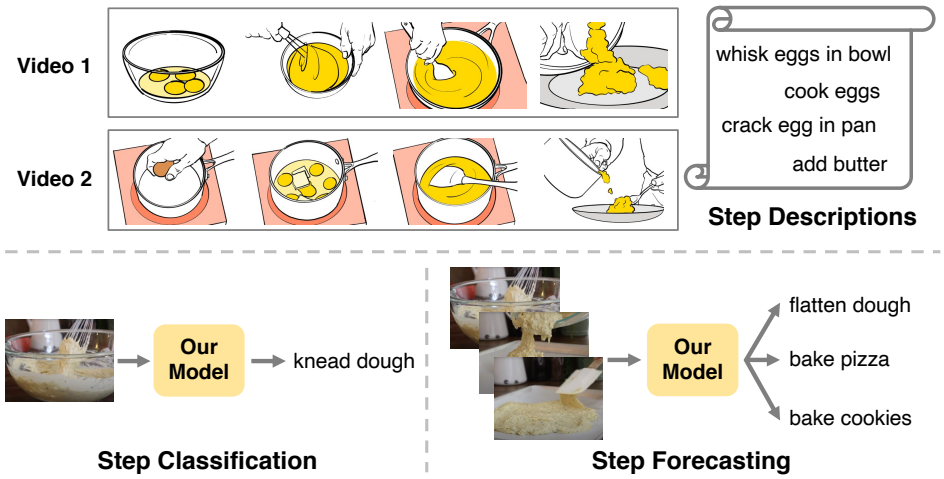


Figure 6.1: **Top**: During training, our model learns from procedural videos and step descriptions to understand individual steps and capture temporal ordering and variations among steps. **Bottom**: Once trained, our model supports zero-shot step classification and forecasting, yielding multiple credible predictions.

their narrations. Our key innovation lies in the joint learning of a video representation aiming to encode individual step concepts, and a deep probabilistic model designed to capture temporal dependencies and variations among steps. The video representation, instantiated as a Transformer network, is learned by matching a video clip to its corresponding narration. The probabilistic model, built on a diffusion process, is tasked to predict the distribution of the video representation for a missing step, given steps in its vicinity. With the help of a pre-trained image-and-language model Radford et al. (2021), our model is trained using only videos and their narrations from automatic speech recognition (ASR), and thus does not require any manual annotations.

Once learned, our model celebrates two unique benefits thanks to our model design and training framework. First, our model supports *zero-shot inference* given an input video, including the recognition of single steps and forecasting of future steps, and can be further fine-tuned on downstream tasks. Second, our model allows *sampling multiple video representations* when predicting a missing action step, with each presenting a possibly different hypothesis of the step ordering. Instead of predicting a single representation with the highest probability, sampling from a probabilistic model provides access to additional high-probability solutions that might be beneficial to prediction tasks with high ambiguity or requiring user interactions.

This chapter is organized as follows. In Section 6.1, I summarize our contributions.

In Section 6.2 and Section 6.3, I present our method and key experiments, respectively. Finally, I conclude the chapter in Section 6.4.

6.1 Contributions

The contributions of this chapter are summarized into three folds:

- Our work presents the first model that leverages video-and-language pre-training to capture the temporal ordering of action steps in procedural activities.
- Our key technical innovation lies in the design of a deep probabilistic model using a diffusion process, in tandem with video-and-language representation learning.
- The result is a model and a training framework that establish new state-of-the-art results on both step classification and forecasting tasks across the major benchmarks. Besides, our model is capable of generating diverse step predictions and supports zero-shot inference.

This work was a collaboration with Licheng Yu, Yang Bai, Shangwen Li, Xueting Yan and Yin Li. The work has been accepted by CVPR 2023 with title as "Learning Procedure-aware Video Representation from Instructional Videos and Their Narrations" Zhong et al. (2023).

6.2 Method

We consider the problem of learning video representation for understanding procedural activities from instructional videos and their narrations. An input video is represented as a sequence of N clips $\{v_1, v_2, ..., v_N\}$. Each v_i captures a potential action step in the input video, and the time step i records the temporal ordering of these clips. The video clips $\{v_i\}$ can be either segmented by using the timestamps of ASR outputs (as we consider during training), or densely sampled from a video following their temporal ordering (as we use during inference). During learning, we further assume that an ordered set of sentences $\{s_1, s_2, ..., s_N\}$ is associated with the video clips $\{v_1, v_2, ..., v_N\}$, with each s_i describing the action step in video clip v_i. These sentences

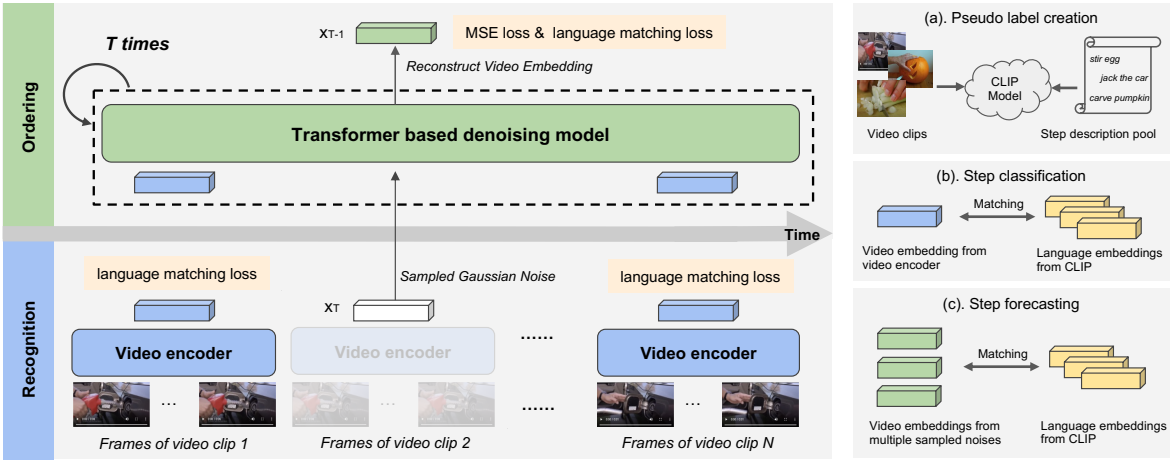


Figure 6.2: Overview of our approach. **Left panel**: Our model consists of (1) a video encoder that takes a video clip and encodes it into a video embedding; (2) a transformer-based denoising model that samples noises from Gaussian distribution and generates video embedddings conditioned on the embeddings of adjacent video clips. **Right panel**: We leverage trained image-language model CLIP to create pseudo labels for individual video clips (a). After training, our model supports step classification given an input video clip (b), and step forecasting given a video that records previous steps (c). Note that diverse embeddings can be generated by sampling various Gaussian noises.

$\{s_i\}$ can be the output text from ASR, or given by matching the video clips to a text corpus using an external vision language model Radford et al. (2021).

Procedural-aware Video Representation. Our goal is to learn video representation that encodes both action step concepts and their temporal dependencies across a range of procedural activities. Our representation consists of (a) a video encoder f that extracts a representation $\mathbf{x}_i$ from an input clip v_i (*i.e.*, $\mathbf{x}_i = f(v_i)$); and (b) a probabilistic model that characterizes the conditional probability $p(\mathbf{x}_j = f(v_j)|\{\mathbf{x}_i = f(v_i)\}_{i\neq j})$ $\forall j$. This design is highly flexible and supports a number of procedural reasoning tasks. f offers video representation suitable to classify individual step in a clip. $p(\mathbf{x}_j|\{\mathbf{x}_i\}_{i\neq j})$ models the temporal dependencies among steps, and can be used to predict the video representation of missing steps and further infer their labels.

Model Overview. To learn our representation, we leverage a pre-trained text encoder g that remains fixed during learning, and extend the idea of masked token modeling, populated in natural language processing Kenton and Toutanova (2019). For each input video and its narrations at training time, we randomly sample a clip v_j from $\{v_1, v_2, ..., v_N\}$ and mask it out. We then train our model to predict the distribution of

$\mathbf{x}_j = f(v_j)$ from $\{\mathbf{x}_i = f(v_i)\}_{i \neq j}$ (*i.e.*, $p(\mathbf{x}_j|\{\mathbf{x}_i\}_{i \neq j})$), align the expectation of the predicted distribution $\mathbb{E}(\mathbf{x}_j)$ with the corresponding text embedding $\mathbf{y}_j = g(s_j)$, and match all other video representations $\{\mathbf{x}_i = f(v_i)\}_{i \neq j}$ to their text embeddings $\{\mathbf{y}_i = g(s_i)\}_{i \neq j}$.

Despite the conceptual similarity, our learning is fundamentally different from masked token prediction. Our method seeks to characterize the distribution of $\mathbf{x}_j$ instead of predicting the most likely $\mathbf{x}_j$, resulting in a more principled approach to capture the temporal dependencies among steps, as well as the new capability of sampling multiple high-probability solutions for $\mathbf{x}_j$. Our method is illustrated in Fig. 6.2. In what follows, we lay out the formulation of our model, and describe its training and inference schemes.

Modeling Action Steps and Their Ordering

Formally, given an input video with its clips $\{v_1, v_2, ..., v_N\}$ and their narrations $\{s_1, s_2, ..., s_N\}$, our method assumes a factorization of $p(\mathbf{Y} = \{\mathbf{y}_i\}|\mathbf{X} = \{\mathbf{x}_i\})$ with video representation $\mathbf{x}_i = f(v_i)$ (learnable) and text embedding $\mathbf{y}_i = f(s_i)$ (pre-trained and fixed).

$$p(\mathbf{Y}|\mathbf{X}) = p(\mathbf{y}_j|\mathbf{x}_j) \cdot p(\mathbf{x}_j|\{\mathbf{x}_i\}_{i \neq j}) \cdot \prod_i p(\mathbf{y}_i|\mathbf{x}_i) \quad \forall j. \tag{6.1}$$

$p(\mathbf{y}_i|\mathbf{x}_i)$ measures the alignment between a video representation $\mathbf{x}_i$ and a text embedding $\mathbf{y}_i$. $p(\mathbf{x}_j|\{\mathbf{x}_i\}_{i \neq j})$ characterizes the distribution of a video representation for a missing step given the representations of all other steps, thereby modeling the temporal ordering of steps. Note that our model is not limited to single step prediction and can be readily extended to predict multiple missing steps.

Matching Image and Text Representations. Our model matches the video representation $\mathbf{x}_i$ and text embedding $\mathbf{y}_i$ in a learned vector space, such that the alignment between them can be measured by cosine similarity. We will later instantiate this definition into a more tractable form for learning. Yet it suffices to notice that $p(\mathbf{y}_j|\mathbf{x}_j)$ does not involve additional learnable parameters given $\mathbf{x}_i$ and $\mathbf{y}_i$.

Modeling Step Ordering with Diffusion Process. The key challenge lies in the modeling of $p(\mathbf{x}_j|\{\mathbf{x}_i\}_{i \neq j})$, as the video representation $\mathbf{x}_i$ is at least of a few hundred dimensions. To this end, we propose to model $p(\mathbf{x}_j|\{\mathbf{x}_i\}_{i \neq j})$ using a diffusion process Sohl-Dickstein et al. (2015); Song and Ermon (2020) conditioned on observed video representations $\{\mathbf{x}_i\}_{i \neq j}$. Here we briefly describe diffusion process in

the context of our model, and refer the readers to recent surveys for more technical details Croitoru et al. (2022); Yang et al. (2022).

Specifically, we assume a diffusion process that gradually adds noise to the input $\mathbf{x}_j$ over $t \in [0, 1, ..., T]$ steps.

$$\begin{aligned} p(\mathbf{x}_j^{1:T}|\mathbf{x}^0) &= \prod_{t=1}^{T} p(\mathbf{x}_j^t|\mathbf{x}_j^{t-1}), \\ p(\mathbf{x}_j^t|\mathbf{x}_j^{t-1}) &= \mathcal{N}(\sqrt{\alpha_t}\mathbf{x}_j^{t-1}, (1-\alpha_t)\mathbf{I}). \end{aligned} \tag{6.2}$$

where α_t are constant hyper-parameters. The reverse diffusion (denoising) process is parameterized with

$$p(\mathbf{x}_j^{0:T}|\{\mathbf{x}_i\}_{i\neq j}) = p(\mathbf{x}_j^T|\{\mathbf{x}_i\}_{i\neq j}) \prod_{t=1}^{T} p_\theta(\mathbf{x}_j^{t-1}|\mathbf{x}_j^t, \{\mathbf{x}_i\}_{i\neq j}). \tag{6.3}$$

With sufficiently large T, $p(\mathbf{x}_j^T|\{\mathbf{x}_i\}_{i\neq j}) \sim \mathcal{N}(0, I)$, *i.e.*a standard Gaussian noise that is independent of $\{\mathbf{x}_i\}_{i\neq j}$. The denoising process is approximated by $p_\theta(\mathbf{x}_j^{t-1}|\mathbf{x}_j^t, \{\mathbf{x}_i\}_{i\neq j})$ using a neural network with parameters θ such that

$$p_\theta(\mathbf{x}_j^{t-1}|\mathbf{x}_j^t, \{\mathbf{x}_i\}_{i\neq j}) = \mathcal{N}(\mu_\theta(\mathbf{x}_j^t, \{\mathbf{x}_i\}_{i\neq j}), \Sigma_\theta(\mathbf{x}_j^t, \{\mathbf{x}_i\}_{i\neq j})) \tag{6.4}$$

In practice, we follow Ho *et al*. Ho et al. (2020) and Tevet *et al*. Tevet et al. (2023) to directly predict $\mathbf{x}_j^0$ by using a denoising model h. With slight abuse of the symbols, we denote

$$\hat{\mathbf{x}}_j^0 = h(\mathbf{x}_j^t, \{\mathbf{x}_i\}_{i\neq j}, t). \tag{6.5}$$

h is realized using a Transformer network with the embedding of step t as part of its inputs. Once learned, one can sample from $\mathcal{N}(0, I)$ and apply h through the denoising process to predict $\mathbf{x}_j$ based on $\{\mathbf{x}_i\}_{i\neq j}$.

Learning from Videos and Their Narrations

Our training approximately maximizes the likelihood of Eq. 6.1 given a set of training videos and their narrations.

Pseudo Labels from CLIP. It is straightforward to directly align video representations to the embeddings of their corresponding ASR text. Doing so, however, faces the

challenges of low-quality ASR text and imprecise alignment between video and ASR sentences. To address these challenges, we propose to create pseudo labels by leveraging a pre-trained image-language model (*e.g.*, CLIP Radford et al. (2021)).

Specifically, we first create a pool of step descriptions in the form of verb phrases (*e.g.*, "add water", "wear gloves") parsed from ASR sentences Shen et al. (2021), with their embeddings as $\{\mathbf{y}^{1:K}\}$. Then a trained CLIP model is applied to link each video clip with verb phrases, by matching the averaged visual features across frames with the language embeddings of verb phrases. The resulting matching scores are used as our training target.

Our pseudo labeling instantiates the matching $p(\mathbf{y}_i|\mathbf{x}_i)$ between video representation and text embedding using

$$p(\mathbf{y}_i|\mathbf{x}_i) = \text{softmax}\left(\frac{\mathbf{x}_i^\top \mathbf{y}_i}{\tau\|\mathbf{x}_i\|\|\mathbf{y}_i\|}\right), \tag{6.6}$$

where $\mathbf{y}_i$ is selected from pool of verb phrases, *i.e.*$\mathbf{y}_i \in \{\mathbf{y}^{1:K}\}$, and τ is the pre-defined temperature. The matching problem thus is converted into a "classification" problem, making the training feasible.

Learning Objective and Training Loss. Our training minimizes an evidence upper bound of the negative log likelihood $-\log p(\mathbf{Y}|\mathbf{X})$ in Eq. 6.1. Our objective function constitutes three loss terms:

$$L = L_{XE} + L_{MSE} + L_{MC}. \tag{6.7}$$

The *first* term L_{XE} seeks to match observed video representation $\{\mathbf{x}_i\}$ to their text embeddings $\{\mathbf{y}_i\}$, given by

$$L_{XE} = \frac{1}{N-1}\sum_i H\left(p'_i, p(\mathbf{y}_i|\mathbf{x}_i)\right), \tag{6.8}$$

where $H(\cdot,\cdot)$ is the cross entropy, and p'_i are soft targets given by CLIP matching scores. $p(\mathbf{y}_i|\mathbf{x}_i)$ defined in Eq. 6.6 measures the similarity between $\mathbf{x}_i$ and $\mathbf{y}_i$.

The *second* term L_{MSE} comes from the Kullback–Leibler (KL) divergence within our diffusion model, and is computed as

$$L_{MSE} = \mathbb{E}_{\mathbf{x}_j^0 \sim p\left(\mathbf{x}_j^0|\{\mathbf{x}_i\}_{i\neq j}\right),\ t\sim[1,T]}\left[\|\mathbf{x}_j^0 - \hat{\mathbf{x}}_j^0\|_2^2\right]. \tag{6.9}$$

Note that unlike a standard diffusion model, our model directly predicts $\hat{\mathbf{x}}_j^0$. This term is applied at each step t.

The *third* term L_{MC} is derived from matching the predicted video representation $\hat{\mathbf{x}}_j^0$ to its text embedding $\mathbf{y}_j$.

$$L_{MC} = \mathbb{E}_{\mathbf{x}_j^0 \sim p\left(\mathbf{x}_j^0 | \{\mathbf{x}_i\}_{i \neq j}\right),\ t \sim [1,T]} \left[H\left(p'_j, p(\mathbf{y}_j | \hat{\mathbf{x}}_j^0)\right) \right], \tag{6.10}$$

where p'_j are again soft targets given by CLIP model, and $\hat{\mathbf{x}}_j^0$ is denoised from a sampled noise. During training, we adopt Monte Carlo estimation for $\mathbb{E}_p \left[-\log p(\mathbf{y}_j | \hat{\mathbf{x}}_j^0) \right]$, by minimizing $-\log p(\mathbf{y}_j | \hat{\mathbf{x}}_j^0)$ for each sampled $\hat{\mathbf{x}}_j^0$. We attach this term at each step t.

A critical design choice lies in $p(\mathbf{y}_j | \hat{\mathbf{x}}_j^0)$. $p(\mathbf{y}_j | \hat{\mathbf{x}}_j^0)$ is simplified into a score function between a video representation and a finite set of text embeddings (defined using verb phrases). This allows us to reach our loss terms without worrying about global normalization constant as commonly encountered in energy-based models. Indeed, $H(p'_j, p(\mathbf{y}_j | \hat{\mathbf{x}}_j^0))$ can be interpreted as providing guidance by matching video to text embeddings. This term thus resembles the key idea of classifier guidance, which has shown to be helpful for learning diffusion models Dhariwal and Nichol (2021).

Model Inference

Once trained, our model offers a procedure-aware representation with two key components. First, the video encoder $f(\cdot)$ serves as a feature extractor for any input video clips. Second, the diffusion model, represented as its denoising model $h(\cdot)$, captures the temporal dependencies among steps. Our representation naturally supports a number of tasks. Here we demonstrate how our model can be used for step classification and step forecasting.

Step Classification. An input video clip v can be encoded using $f(\cdot)$. The video representation $\mathbf{x} = f(v)$ can be directly compared to the text embeddings Radford et al. (2021), so as to support zero-shot step classification. Alternative, an additional classifier can be attached on top of f and further fine-tuned to recognize the action step in input clip.

Step Forecasting. A future video clip feature $\mathbf{x}_j$ can be sampled from the diffusion model by drawing from a Gaussian distribution and denoising using $h(\cdot)$. The predicted $\mathbf{x}_j$ can be further classified into action steps. This prediction can be done using

again Monte Carlo estimation given by

$$\mathbb{E}_{\mathbf{x}_j \sim p(\mathbf{x}_j | \{\mathbf{x}_i\}_{i \neq j})} \left[p(\mathbf{y}_j | \mathbf{x}_j) \right]. \tag{6.11}$$

Specifically, a noise $\mathbf{x}_T$ is first sampled from Gaussian distribution and our denoising model gradually denoises it. At each step t, the denoising model h takes a noisy $\mathbf{x}_j^t$, predicts clip feature $\hat{\mathbf{x}}_j^0$, and diffuses it to $\mathbf{x}_j^{t-1}$ based on the sampled noise $\mathbf{x}_j^T$, as demonstrated in Eq. 6.2. After T iterations, the predicted clip feature at $t = 0$ is used to match text embeddings. By sampling noises for multiple times, we can estimate the most likely $\mathbf{y}_j$.

However, sampling can be costly. In practice, to obtain top-1 prediction for missing steps, we adopt approximate inference, where the sampled noise is replaced with a fixed zero vector, corresponding to peak in the Gaussian distribution. Our empirical results validate that approximate inference achieves a very close performance as the expectation over multiple sampled noises.

6.3 Key Experiments

In this section, we first introduce datasets, evaluation protocols and implementation details. Then we demonstrate our results on step forecasting and step classification benchmarks. Finally, we show our qualitative results.

Datasets. For *pre-training*, we consider HowTo100M dataset Miech et al. (2019) with 130K hours of YouTube tutorial videos. The videos cover various daily tasks, such as foods, housework, vehicles, and so on. We use a language parser Shen et al. (2021) to extract the verb phrases from ASR sentences of these videos and keep 9,871 most frequent verb phrases. For *fine-tuning*, we train our model on COIN dataset Tang et al. (2019b, 2020b) and EPIC-Kitchens-100 dataset Damen et al. (2021), respectively. COIN has 476 hours of YouTube videos covering 180 tasks, such as dishes, vehicles, and housework. Human annotators summarize 778 unique steps in total (*e.g.*, "stir the egg"), and annotate the temporal boundary and the category of each step in all videos. EPIC-Kitchens-100 dataset Damen et al. (2021) has 100 hours of egocentric videos, capturing daily activities in kitchen. Each action in the videos is annotated with an action label and a noun label. There are 97/300 unique actions/nouns in total. We use the human annotations in COIN and EPIC-Kitchens-100 for model evaluation.

Evaluation Protocols. Our evaluation considers zero-shot and fine-tuning settings for step classification and step forecasting on COIN and EPIC-Kitchens-100 datasets. Zero-shot setting indicates that no human annotation is used during pre-training. The pre-trained model is directly tested on the evaluation dataset. Fine-tuning setting further fine-tunes the pre-trained model using human annotations of action steps. For a fair comparison, we follow the same fine-tuning schemes as previous work in respective benchmarks.

Implementation Details. We adopted TimeSformer Bertasius et al. (2021) as our video encoder, and used the Transformer Vaswani et al. (2017b) from CLIP's text encoder as denoising model. We set the maximum step T to 4, maximum length of video sequence as 9, and the number of Transformer layers as 4. We used a trained CLIP model (ViT-B/16) to create pseudo labels and encode step descriptions. Following DistantSup Lin et al. (2022), for pre-training we used SGD for 5 epochs and then AdamW Loshchilov and Hutter (2018) for 25 epochs with 128 videos in a batch. For fine-tuning, we used AdamW for 15 epochs with batch size of 64. Temperature τ was set to 0.02.

Step Forecasting

Setup. We follow the benchmark in DistantSup Lin et al. (2022) to evaluate step forecasting on COIN, where top-1 accuracy is reported. Given a video with previous steps, the model anticipates the category of next single step (*e.g.*, "stir the egg"). This task thus requires explicit modeling of the temporal ordering among steps. We only fine-tune the diffusion model while keeping the video encoder frozen.

Results. Table 6.1 compares results of our method with a series of baselines. The closest competitor is DistantSup Lin et al. (2022) in L5, which learns from ASR text and an external textual knowledge base Koupaee and Wang (2018) using the same video backbone (TimeSformer Bertasius et al. (2021)). We also include other baselines reported in DistantSup, *e.g.*, SlowFast Feichtenhofer et al. (2019), TimeSformer Bertasius et al. (2021), and S3D Xie et al. (2018) from L1 to L4, where models are supervised using human-annotated action labels or video ASR text. Our model significantly outperforms all baselines by at least **7.4%** for the fine-tuning setting (*e.g.*, 46.8% in L8 vs. 39.4% in L5). Further, we consider a strong baseline for the zero-shot setting by re-purposing CLIP model Radford et al. (2021) to match the input video with the descriptions of all step candidates. Comparing L7 and L8, our model outperforms

	Model	Pretraining		Top-1 Acc. (%)	
		Supervision	Dataset	Zero-shot	Fine-tuning
1	SlowFast Feichtenhofer et al. (2019)	Supervised: action labels	Kinetics	–	25.6
2	TimeSformer Bertasius et al. (2021)	Supervised: action labels	Kinetics	–	34.7
3	S3D Xie et al. (2018)	Unsupervised: ASR w. MIL-NCE Miech et al. (2020)	HT100M	–	28.1
4	TimeSformer Bertasius et al. (2021)	Unsupervised: ASR w. MIL-NCE Miech et al. (2020)	HT100M	–	34.0
5	DistantSup Lin et al. (2022)	Unsupervised: ASR + wikiHow	HT100M	–	39.4
6	Random Guess	–	–	0.1	–
7	CLIP Radford et al. (2021)	Unsupervised: captions	CLIP400M	9.4	–
8	Ours	Unsupervised: ASR	HT100M	**11.3**	**46.8**
9	Ours (oracle-5)	Unsupervised: ASR	HT100M	14.7	51.8

Table 6.1: Step forecasting on COIN dataset. We compare to a set of strong baselines and a oracle protocol built on our method.

this variant of CLIP by a clear margin (11.3% vs. 9.4%).

A unique property of our model is its ability to output multiple, potentially different predictions. We further evaluate the upper bound of our results by assuming an oracle ranking function that always selects the correction prediction from 5 outputs sampled from our model (Ours (oracle-5)). This oracle further improves the top-1 accuracy from 11.3% to 14.7% in L9, suggesting that our model is able to produce diverse predictions for step forecasting.

Step Classification

Setup. Besides step ordering, we also evaluate step classification on COIN and EPIC-Kitchens-100 datasets, where a model is tasked to classify a trimmed video clip into one of the step categories. For COIN, we follow DistantSup Lin et al. (2022) to only fine-tune the additional linear layer on top of the pre-trained video encoder. For EPIC-Kitchens-100, we fully fine-tune the video encoder, following Kondratyuk et al. (2021); Arnab et al. (2021); Lin et al. (2022). We report the accuracy of step classification on COIN, and that of verb, noun, and action on EPIC-Kitchens-100.

Results. Table. 6.2 summarizes the results on COIN. We consider baselines as in DistantSup from L1 to L8 (*e.g.*, SlowFast Feichtenhofer et al. (2019), VideoCLIP Xu et al. (2021a)), where models are trained using either action labels or video ASR text. To support zero-shot inference, we re-implement a model variant (DistantSup†) described in Lin et al. (2022). This model is pre-trained to match video embeddings with language embeddings and thus supports recognizing arbitrary step descriptions in L9. We also report the results of CLIP Radford et al. (2021), which creates the pseudo labels for our pre-training in L10. As shown, our model consistently outper-

	Model	Pretraining		Top-1 Acc. (%)	
		Supervision	Dataset	Zero-shot	Fine-tuning
1	TSN (RGB+Flow) Tang et al. (2019b)	Supervised: action labels	Kinetics	–	36.5*
2	S3D Xie et al. (2018)	Unsupervised: ASR w. MIL-NCE Miech et al. (2020)	HT100M	–	37.5*
3	SlowFast Feichtenhofer et al. (2019)	Supervised: action labels	Kinetics	–	32.9
4	TimeSformer Bertasius et al. (2021)	Supervised: action labels	Kinetics	–	48.3
5	ClipBERT Lei et al. (2021)	Supervised: captions	COCO+VG	–	30.8
6	VideoCLIP Xu et al. (2021a)	Unsupervised: ASR	HT100M	–	39.4
7	TimeSformer Bertasius et al. (2021)	Unsupervised: ASR w. MIL-NCE Miech et al. (2020)	HT100M	–	46.5
8	DistantSup Lin et al. (2022)	Unsupervised: ASR + wikiHow	HT100M	–	54.1
9	DistantSup† Lin et al. (2022)	Unsupervised: ASR + wikiHow	HT100M	10.2	46.6
10	CLIP Radford et al. (2021)	Unsupervised: captions	CLIP400M	14.8	45.9
11	Ours	Unsupervised: ASR	HT100M	**16.6**	**56.9**

Table 6.2: Step classification on COIN dataset. DistantSup† is re-implemented based on their official code base. It is a variant reported in their paper that pre-trains the model to match language embeddings. * indicates the model is fully fine-tuned.

	Model	Pretraining Supervision	Pretraining Dataset	Action (%)	Verb (%)	Noun (%)
1	TSN Wang et al. (2016)	–	–	33.2	60.2	46.0
2	TRN Zhou et al. (2018a)	–	–	35.3	65.9	45.4
3	TBN Kazakos et al. (2019)	–	–	36.7	66.0	47.2
4	MoViNet Kondratyuk et al. (2021)	–	–	**47.7**	**72.2**	57.3
5	TSM Lin et al. (2019)	Supervised: action labels	Kinetics	38.3	67.9	49.0
6	SlowFast Feichtenhofer et al. (2019)	Supervised: action labels	Kinetics	38.5	65.6	50.0
7	ViViT-L Arnab et al. (2021)	Supervised: action labels	Kinetics	44.0	66.4	56.8
8	TimeSformer Bertasius et al. (2021)	Supervised: action labels	Kinetics	42.3	66.6	54.4
9	DistantSup Lin et al. (2022)	Unsupervised: ASR + wikiHow	HT100M	44.4	67.1	58.1
10	Ours	Unsupervised: ASR	HT100M	**47.7**	69.5	**60.3**

Table 6.3: Step classification on EPIC-Kitchens-100 dataset with fine-tuning setting. Our method outperforms the close competitors (TimeSformer, DistantSup), with results on par with even stronger backbone models (MoViNet).

forms all the other methods by a clear margin under different settings. For example, ours outperforms CLIP by **1.8%** in zero-shot setting (16.6% in L11 vs. 14.8% in L10), and outperforms DistantSup by **2.8%** (56.9% in L11 vs. 54.1% in L8) in fine-tuning.

Table. 6.3 presents our results on EPIC-Kitchens-100. While TimeSformer Bertasius et al. (2021) in L8 and DistantSup Lin et al. (2022) in L9 use the same video encoder architecture as ours, our model in L10 achieves a clear gain over them, *e.g.*, **+3.3%/2.2%** for action/noun. The only exception is the lower accuracy (-2.7%) on verb when compared with MoViNet (MoViNet-A6) in L4, a heavily optimized video backbone.

Predicting Diverse Future Steps

One of the defining characteristics of our model is that it allows us to sample multiple predictions of video representation corresponding to a future step. This leads to an interesting question about the diversity of the predictions, as partially evaluated in our prior experiments. Here we present further demonstration of this capability by

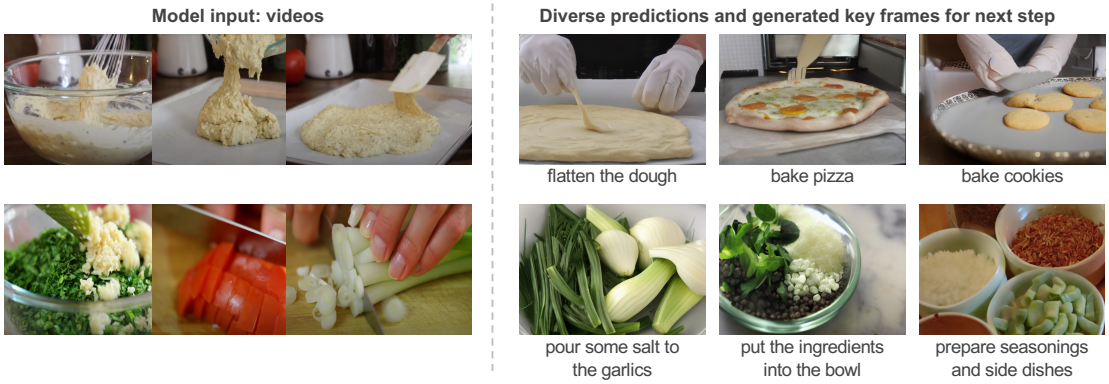


Figure 6.3: Visualization of **zero-shot** step forecasting and **key frame generation**. Without using any human annotation during training, our trained model is directly evaluated on COIN dataset Tang et al. (2020b). Given a video recording previous steps (left), our model is capable of forecasting multiple reasonable predictions and each predicted step is further used for key frame generation (right). We adopt stable diffusion Rombach et al. (2022) for key frame generation, taking inputs as a text description of step and a sampled frame from input video.

visualizing the step forecasting results, and more interestingly, using these results to generate future video frames.

Fig. 6.3 presents the visualization for zero-shot step forecasting and key frame generation. In this setting, our model is pre-trained without any human annotation and is directly tested for step forecasting. We show multiple predictions sampled from our diffusion model. Further, we demonstrate that the text description of predicted step can be used to generate the key frames by leveraging the stable diffusion model Rombach et al. (2022). To keep the generated images visually consistent with the input video, we let stable diffusion model take one input video frame and the description of predicted step as input and generate an image.

As shown in Fig. 6.3, our model can forecast multiple, reasonable next steps (*e.g.*, "flatten the dough", "bake pizza"), based on which credible future frames are generated. These results suggest that our model not only predicts meaningful video representations of individual steps, but also captures the variations in step ordering.

6.4 Conclusion

In this chapter, we presented a model and a training frame work for learning procedure-aware video representation from a large-scale dataset of instructional videos and

their narrations, without the need for human annotations. The key strength of our model lies in the joint learning of a video encoder capturing concepts of action steps, as well as a diffusion model reasoning about the temporal dependencies among steps. We demonstrated that our model achieves strong results on step classification and forecasting in both zero-shot and fine-tuning settings and across COIN and EPIC-Kitchens-100 datasets. We believe our work provides a solid step towards understanding procedural activities. We hope that our work will shed light on the broader problem of video-language pre-training.

7 CONCLUSIONS AND FUTURE WORK

My previous research has extensively explored the methodology of acquiring diverse visual knowledge through human language supervision, utilizing visual-text pairs as the sole training data. This paradigm has proven effective in creating vision models that exhibit generalizability in many visual concepts across a broad spectrum of vision tasks, thus advancing the understanding of the open visual world. Building upon this foundation, I am motivated to broaden the scope of my research toward multi-modal understanding, with the ultimate goal as artificial general intelligence (AGI). This broader research agenda shares the same spirit of learning visual knowledge from human language. In this final chapter, I summarize the contributions and discuss my future research agenda.

7.1 Conclusions

My dissertation, for the first time, demonstrates that learning from visual-text pairs can build vision models that are generalizable to diverse visual concepts in broad vision tasks, and capable of understanding the open visual world. By using the visual-text pairs as the only training data, a spectrum of visual knowledge can be learned, including object concepts, object relationships, scene components, and action procedures. More importantly, open-vocabulary recognition is enabled for each aspect of visual knowledge, without the need for exhaustive human annotations used by fully-supervised methods. I revisit my key contributions as follows:

- My work on object concept learning demonstrated that fine-grained region-token alignment can significantly improve the quality of region representation during image-language pretraining, with image-text pairs as only training data.
- My work on object relationship learning for the first time, validated that high-quality scene graphs can be learned from image-text pairs without the need for human annotation.
- My work on scene component learning showed for the first time that multiple captioning capabilities can be unified in a single model, by learning to select salient sub-graphs from scene graphs.

- My work on action procedure learning for the first time captured the action ordering and its immense variations from unannotated videos and further improved the learning of video representation during video-language pretraining.

My research also extended to the problems of grounding image regions to human-annotated scene graphs Shi et al. (2021a) and to phrases in sentences Li et al. (2022), and the problem of building interpretable visual recognition models explained by language-described visual attributes Yan et al. (2023).

7.2 Future Work

Language supervision from large language models (LLMs). Recently, LLMs Raffel et al. (2020); Chung et al. (2022); Kojima et al. (2022); Touvron et al. (2023) have witnessed unprecedented advancements and have achieved human-level comprehension of text, such as multi-round conversations (*e.g.*, ChatGPT). These models leverage vast textual datasets, align responses with human values via reinforcement learning, and showcase impressive zero-shot and few-shot reasoning capabilities. Given the human-like proficiency of these chatbots, a natural question arises: *Can LLMs be employed to acquire a broader range of visual knowledge beyond visual-text pairs from the web?* Early open-source efforts have emerged, aiming to empower LLMs with the ability to understand visual inputs through techniques like visual instruction tuning (e.g., InstructBLIP Dai et al. (2023), miniGPT4 Zhu et al. (2023), LLaVA Liu et al. (2023)). These methods typically collect instruction-answer pairs and gear the pre-trained LLMs to generate answers, with inputs as instruction and an image. However, these efforts heavily incline on how to generate human-like answers while paying less attention to how to extract and understand rich information from images. For instance, images are typically downsampled to a low resolution and encoded into image features via a pre-trained image encoder, leading to an inevitable loss of fine-grained visual details (*e.g.*, how exactly each object looks like, how these objects interact with each other in the scene). Hence, the open research question that remains is how to exploit LLMs for visual-conditioned understanding, reasoning, and planning, while comprehensively preserving the richness of visual information.

Language supervision from human-robot interaction. Moving beyond large language models, learning visual knowledge from human-robot interaction offers a paradigm that closely resembles how humans learn about the world. By gradually

acquiring new concepts and correcting misconceptions through human instructions, AI models can evolve and align themselves with human intelligence and values. However, relying on newly-generated human instruction is currently impractical due to the high cost associated with setting up human-robot interaction environments and collecting data. The existing efforts in embodied AI focus on creating simulated environments Savva et al. (2019); Wani et al. (2020); Weihs et al. (2021) where the models are trained to navigate rooms and to localize target objects. They are still far from the ability to interact with humans and the real daily-life environment, not to mention achieving artificial general intelligence (AGI). Therefore, this research direction holds significant importance and will be long-lasting in the field.

The benefits of visual symbolic representation. One of the fundamental distinctions between visual and textual data lies in their respective information densities. Human language has been created through a few thousand tokens that form textual data. These textual data already exist as symbolic representations. On the other hand, visual data consists of continuous signals that are subsequently converted into digital pixels. A single image already includes millions of pixels. As a result, visual data is inherently more complex than textual data, containing much richer information. An illustration of this complexity is the difficulty faced by a layperson drawing a mimic picture solely based on the text description. In light of the achievements of large language models (LLMs), an intriguing question arises: *can visual symbolic representation facilitate visual understanding?* Scene graphs Krishna et al. (2017); Xu et al. (2017), serving as abstract and symbolic representations, can be seen as a tokenization process applied to the visual world. In my previous work Zhong et al. (2020), I have demonstrated the advantages of scene graphs in controllable and grounded captioning tasks. It is worthwhile to further explore the potential benefits of visual symbolic representation in advancing visual understanding and robot manipulation.

REFERENCES

Alayrac, Jean-Baptiste, Piotr Bojanowski, Nishant Agrawal, Josef Sivic, Ivan Laptev, and Simon Lacoste-Julien. 2016. Unsupervised learning from narrated instruction videos. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 4575–4583.

Anderson, Peter, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *Proceedings of the european conference on computer vision (eccv)*, ed. Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, 382–398. Springer.

Anderson, Peter, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 6077–6086. IEEE.

Aneja, Jyoti, Harsh Agrawal, Dhruv Batra, and Alexander Schwing. 2019. Sequential latent spaces for modeling the intention during diverse image captioning. In *Proceedings of the ieee international conference on computer vision (iccv)*, 4261–4270. IEEE.

Armeni, Iro, Zhi-Yang He, JunYoung Gwak, Amir R Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 2019. 3D scene graph: A structure for unified semantics, 3d space, and camera. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 5664–5673.

Arnab, Anurag, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. 2021. Vivit: A video vision transformer. In *Proceedings of the ieee/cvf international conference on computer vision*, 6836–6846.

Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Bain, Max, Arsha Nagrani, Gül Varol, and Andrew Zisserman. 2021. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the ieee/cvf international conference on computer vision*, 1728–1738.

Baldassarre, Federico, Kevin Smith, Josephine Sullivan, and Hossein Azizpour. 2020. Explanation-based weakly-supervised learning of visual relations with graph

networks. In *Proceedings of the european conference on computer vision (eccv)*, 612–630. Springer.

Banerjee, Satanjeev, and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, 65–72.

Bansal, Ankan, Karan Sikka, Gaurav Sharma, Rama Chellappa, and Ajay Divakaran. 2018. Zero-shot object detection. In *Eccv*, 384–400.

Barnard, Kobus, Pinar Duygulu, David Forsyth, Nando de Freitas, David M. Blei, and Michael I. Jordan. 2003. Matching words and pictures. *Journal of machine learning research* 3:1107–1135.

Bertasius, Gedas, Heng Wang, and Lorenzo Torresani. 2021. Is space-time attention all you need for video understanding? In *Proceedings of the international conference on machine learning (icml)*.

Bird, Steven, and Edward Loper. 2004. NLTK: The natural language toolkit. In *ACL interactive poster and demonstration sessions*, 214–217. Association for Computational Linguistics.

Bojanowski, Piotr, Rémi Lajugie, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. 2014. Weakly supervised action labeling in videos under ordering constraints. In *European conference on computer vision*, 628–643. Springer.

Bojanowski, Piotr, Rémi Lajugie, Edouard Grave, Francis Bach, Ivan Laptev, Jean Ponce, and Cordelia Schmid. 2015. Weakly-supervised alignment of video with text. In *Proceedings of the ieee international conference on computer vision*, 4462–4470.

Brand, M., N. Oliver, and A. Pentland. 1997. Coupled hidden markov models for complex action recognition. In *Proceedings of ieee computer society conference on computer vision and pattern recognition*, 994–999.

Carion, Nicolas, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *Eccv*, 213–229. Springer.

Caron, Mathilde, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*.

Chang, Chien-Yi, De-An Huang, Danfei Xu, Ehsan Adeli, Li Fei-Fei, and Juan Carlos Niebles. 2020. Procedure planning in instructional videos. In *European conference on computer vision*, 334–350. Springer.

Chen, Long, Hanwang Zhang, Jun Xiao, Xiangnan He, Shiliang Pu, and Shih-Fu Chang. 2019a. Counterfactual critic multi-agent training for scene graph generation. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 4613–4623.

Chen, Tianshui, Weihao Yu, Riquan Chen, and Liang Lin. 2019b. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 6163–6171.

Chen, Ting, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020a. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Chen, Xinlei, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. 2015. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*.

Chen, Xinlei, and Abhinav Gupta. 2015. Webly supervised learning of convolutional networks. In *Proceedings of the ieee international conference on computer vision (iccv)*, 1431–1439.

Chen, Xinlei, Abhinav Shrivastava, and Abhinav Gupta. 2013. NEIL: Extracting visual knowledge from web data. In *Proceedings of the ieee international conference on computer vision (iccv)*, 1409–1416.

Chen, Yen-Chun, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020b. UNITER: Universal image-text representation learning. In *Eccv*, 104–120. Springer.

Chung, Hyung Won, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Cornia, Marcella, Lorenzo Baraldi, and Rita Cucchiara. 2019. Show, control and tell: A framework for generating controllable and grounded captions. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 8307–8316. IEEE.

Croitoru, Florinel-Alin, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. 2022. Diffusion models in vision: A survey. *arXiv preprint arXiv:2209.04747*.

Dai, Bo, Sanja Fidler, Raquel Urtasun, and Dahua Lin. 2017a. Towards diverse and natural image descriptions via a conditional gan. In *Proceedings of the ieee international conference on computer vision (iccv)*, 2970–2979. IEEE.

Dai, Bo, Yuqi Zhang, and Dahua Lin. 2017b. Detecting visual relationships with deep relational networks. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 3076–3086.

Dai, Wenliang, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. 2023. Instructblip: Towards general-purpose vision-language models with instruction tuning. *arXiv preprint arXiv:2305.06500*.

Damen, Dima, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. 2021. The EPIC-KITCHENS dataset: Collection, challenges and baselines. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 43(11):4125–4141.

Das, Abhishek, Harsh Agrawal, Larry Zitnick, Devi Parikh, and Dhruv Batra. 2017. Human attention in visual question answering: Do humans and deep networks look at the same regions? *Computer Vision and Image Understanding* 163:90–100.

Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 248–255. IEEE.

Desai, Karan, and Justin Johnson. 2021. VirTex: Learning visual representations from textual annotations. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*.

Deshpande, Aditya, Jyoti Aneja, Liwei Wang, Alexander G Schwing, and David Forsyth. 2019. Fast, diverse and accurate image captioning guided by part-of-speech. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 10695–10704. IEEE.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics (naacl)*, 4171–4186. Minneapolis, Minnesota: Association for Computational Linguistics.

Dhamo, Helisa, Azade Farshad, Iro Laina, Nassir Navab, Gregory D. Hager, Federico Tombari, and Christian Rupprecht. 2020. Semantic image manipulation using scene graphs. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*.

Dhariwal, Prafulla, and Alexander Nichol. 2021. Diffusion models beat GANs on image synthesis. *Advances in Neural Information Processing Systems* 34:8780–8794.

Divvala, Santosh K, Ali Farhadi, and Carlos Guestrin. 2014. Learning everything about anything: Webly-supervised visual concept learning. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 3270–3277.

Donahue, Jeffrey, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 2625–2634. IEEE.

Dosovitskiy, Alexey, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International conference on learning representations (iclr)*.

Elhamifar, E., and D. Huynh. 2020. Self-supervised multi-task procedure learning from instructional videos. *European Conference on Computer Vision*.

Elhamifar, Ehsan, and Zwe Naing. 2019. Unsupervised procedure learning via joint dynamic summarization. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*.

Everingham, Mark, Luc Gool, Christopher K. Williams, John Winn, and Andrew Zisserman. 2010. The pascal visual object classes (voc) challenge. *IJCV*.

Fan, Angela, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th annual meeting of the association for computational linguistics (acl)*, 889–898. Association for Computational Linguistics.

Farhadi, Ali, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 1778–1785. IEEE.

Feichtenhofer, Christoph, Haoqi Fan, Jitendra Malik, and Kaiming He. 2019. SlowFast networks for video recognition. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*.

Felzenszwalb, Pedro F, and Daniel P Huttenlocher. 2004. Efficient graph-based image segmentation. *International Journal of Computer Vision (IJCV)* 59(2):167–181.

Fu, Tsu-Jui, Linjie Li, Zhe Gan, Kevin Lin, William Yang Wang, Lijuan Wang, and Zicheng Liu. 2021. VIOLET: End-to-end video-language transformers with masked visual-token modeling. *arXiv preprint arXiv:2111.12681*.

Ghadiyaram, Deepti, Du Tran, and Dhruv Mahajan. 2019. Large-scale weakly-supervised pre-training for video action recognition. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 12046–12055.

Ghiasi, Golnaz, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. 2021. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 2918–2928.

Girshick, Ross, Jeff Donahue, Trevor Darrell, and Jitendra Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 580–587.

Goel, Karan, and Emma Brunskill. 2019. Learning procedural abstractions and evaluating discrete latent temporal structure. In *International conference on learning representations*.

Grill, Jean-Bastien, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. 2020. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*.

Gu, Xiuye, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. 2021. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*.

Gupta, Abhinav, Praveen Srinivasan, Jianbo Shi, and Larry S. Davis. 2009. Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In *2009 ieee conference on computer vision and pattern recognition*, 2012–2019.

Gupta, Agrim, Piotr Dollar, and Ross Girshick. 2019. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 5356–5364.

Han, Tengda, Weidi Xie, and Andrew Zisserman. 2022. Temporal alignment networks for long-term video. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*.

He, Kaiming, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 9729–9738.

He, Kaiming, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the ieee international conference on computer vision*, 2961–2969.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 770–778.

He, Xiangteng, and Yuxin Peng. 2017. Fine-grained image classification via combining vision and language. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 5994–6002.

Hénaff, Olivier J., Skanda Koppula, Jean-Baptiste Alayrac, Aaron van den Oord, Oriol Vinyals, and João Carreira. 2021. Efficient visual pretraining with contrastive

detection. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 10086–10096.

Hironobu, Yasuhide Mori, Hironobu Takahashi, and Ryuichi Oka. 1999. Image-to-word transformation based on dividing and vector quantizing images with words. In *in boltzmann machines", neural networks*, 405409.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33:6840–6851.

Ho, Jonathan, Tim Salimans, Alexey A Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. 2022. Video diffusion models. In *Advances in neural information processing systems*.

Hossain, MD Zakir, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. 2019. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CSUR)* 51(6):1–36.

Hudson, Drew, and Christopher D Manning. 2019. Learning by abstraction: The neural state machine. In *Advances in neural information processing systems (neurips)*, vol. 32. Curran Associates, Inc.

Ivanov, Y.A., and A.F. Bobick. 2000. Recognition of visual activities and interactions by stochastic parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(8):852–872.

Jerbi, Achiya, Roei Herzig, Jonathan Berant, Gal Chechik, and Amir Globerson. 2020. Learning object detection from captions via textual scene attributes. *arXiv preprint arXiv:2009.14558*.

Ji, Jingwei, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. 2020. Action Genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 10236–10247.

Jia, Chao, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V Le, Yunhsuan Sung, Zhen Li, and Tom Duerig. 2021. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning (icml)*.

Jianbo Shi, and J. Malik. 2000. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (*TPAMI*) 22(8):888–905.

Jiayuan, Mao, and Kasai Seito. 2018. Scene graph parser. *https://github.com/vacancy/SceneGraphParser*.

Johnson, Justin, Agrim Gupta, and Li Fei-Fei. 2018. Image generation from scene graphs. In *Proceedings of the ieee conference on computer vision and pattern recognition* (*cvpr*).

Johnson, Justin, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the ieee conference on computer vision and pattern recognition* (*cvpr*), 4565–4574. IEEE.

Joulin, Armand, Laurens Van Der Maaten, Allan Jabri, and Nicolas Vasilache. 2016. Learning visual features from large weakly supervised data. In *Eccv*, 67–84. Springer.

Karpathy, Andrej, and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the ieee conference on computer vision and pattern recognition* (*cvpr*), 3128–3137. IEEE.

Kazakos, Evangelos, Arsha Nagrani, Andrew Zisserman, and Dima Damen. 2019. EPIC-fusion: Audio-visual temporal binding for egocentric action recognition. In *Proceedings of the ieee/cvf international conference on computer vision*, 5492–5501.

Kenton, Jacob Devlin Ming-Wei Chang, and Lee Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacl-hlt*, 4171–4186.

Kim, Dong-Jin, Jinsoo Choi, Tae-Hyun Oh, and In So Kweon. 2019. Dense relational captioning: Triple-stream networks for relationship-based captioning. In *Proceedings of the ieee conference on computer vision and pattern recognition* (*cvpr*), 6271–6280. IEEE.

Kingma, Diederik P, and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International conference on learning representations* (*iclr*).

Kipf, Thomas N, and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. In *International conference on learning representations* (*iclr*).

Klusowski, Jason M., and Yihong Wu. 2018. Counting motifs with graph sampling. In *Colt*, 1966–2011. Proceedings of Machine Learning Research.

Kojima, Takeshi, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems* 35:22199–22213.

Kondratyuk, Dan, Liangzhe Yuan, Yandong Li, Li Zhang, Mingxing Tan, Matthew Brown, and Boqing Gong. 2021. MoViNets: Mobile video networks for efficient video recognition. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 16020–16030.

Koupaee, Mahnaz, and William Yang Wang. 2018. WikiHow: A large scale text summarization dataset. *arXiv preprint arXiv:1810.09305*.

Krishna, Ranjay, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV)* 123(1):32–73.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, vol. 25.

Kuehne, Hilde, Ali Arslan, and Thomas Serre. 2014. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*.

Kukleva, Anna, Hilde Kuehne, Fadime Sener, and Jurgen Gall. 2019. Unsupervised learning of action classes with continuous temporal embedding. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 12066–12074.

Kuznetsova, Alina, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. 2020. The open images dataset v4. *International Journal of Computer Vision (IJCV)* 1–26.

Lampert, Christoph H, Hannes Nickisch, and Stefan Harmeling. 2009. Learning to detect unseen object classes by between-class attribute transfer. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 951–958. IEEE.

Leacock, Claudia, George A Miller, and Martin Chodorow. 1998. Using corpus statistics and wordnet relations for sense identification. *Computational Linguistics* 24(1):147–165.

Lei, Jie, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is more: Clipbert for video-and-language learning via sparse sampling. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 7331–7341.

Li, Dianqi, Qiuyuan Huang, Xiaodong He, Lei Zhang, and Ming-Ting Sun. 2018a. Generating diverse and accurate visual captions by comparative adversarial learning. *arXiv preprint arXiv:1804.00861*.

Li, Junnan, Ramprasaath R Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *arXiv preprint arXiv:2107.07651*.

Li, Linjie, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. 2020a. HERO: Hierarchical encoder for Video+Language omni-representation pre-training. In *Proceedings of the 2020 conference on empirical methods in natural language processing (emnlp)*, 2046–2065. Online: Association for Computational Linguistics.

Li, Liunian Harold, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. 2022. Grounded language-image pre-training. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*.

Li, Xiujun, Xi Yin, Chunyuan Li, Pengchuan Zhang, Xiaowei Hu, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, et al. 2020b. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *Eccv*, 121–137. Springer.

Li, Yikang, Wanli Ouyang, Bolei Zhou, Jianping Shi, Chao Zhang, and Xiaogang Wang. 2018b. Factorizable net: an efficient subgraph-based framework for scene graph generation. In *Proceedings of the european conference on computer vision (eccv)*, 335–351.

Li, Yikang, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. 2017. Scene graph generation from objects, phrases and region captions. In *Proceedings of the ieee international conference on computer vision (iccv)*, 1261–1270.

Lin, Chin-Yew. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, 74–81.

Lin, Ji, Chuang Gan, and Song Han. 2019. TSM: Temporal shift module for efficient video understanding. In *Proceedings of the ieee/cvf international conference on computer vision*, 7083–7093.

Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Eccv*, 740–755. Springer.

Lin, Xudong, Fabio Petroni, Gedas Bertasius, Marcus Rohrbach, Shih-Fu Chang, and Lorenzo Torresani. 2022. Learning to recognize procedural activities with distant supervision. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 13853–13863.

Liu, Fenglin, Xuancheng Ren, Yuanxin Liu, Houfeng Wang, and Xu Sun. 2018. simNet: Stepwise image-topic merging network for generating detailed and comprehensive image captions. In *Proceedings of the 2018 conference on empirical methods in natural language processing (emnlp)*, 137–149. Association for Computational Linguistics.

Liu, Haotian, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Loshchilov, Ilya, and Frank Hutter. 2018. Decoupled weight decay regularization. In *International conference on learning representations*.

Lu, Cewu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. 2016. Visual relationship detection with language priors. In *Proceedings of the european conference on computer vision (eccv)*, 852–869. Springer.

Lu, Jiasen, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *Advances in neural information processing systems (neurips)*, 13–23.

Lu, Jiasen, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 2020. 12-in-1: Multi-task vision and language representation learning. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*.

Lu, Jiasen, Caiming Xiong, Devi Parikh, and Richard Socher. 2017. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 375–383. IEEE.

Lu, Jiasen, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2018. Neural baby talk. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 7219–7228. IEEE.

Luo, Huaishao, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. 2020. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*.

Luo, Ruotian, and Gregory Shakhnarovich. 2020. Analysis of diversity-accuracy tradeoff in image captioning. *arXiv preprint arXiv:2002.11848*.

Ma, Chih-Yao, Yannis Kalantidis, Ghassan AlRegib, Peter Vajda, Marcus Rohrbach, and Zsolt Kira. 2019. Learning to generate grounded image captions without localization supervision. *arXiv preprint arXiv:1906.00283*.

Malmaud, Jonathan, Jonathan Huang, Vivek Rathod, Nick Johnston, Andrew Rabinovich, and Kevin Murphy. 2015. What's cookin'? interpreting cooking videos using text, speech and vision. *arXiv preprint arXiv:1503.01558*.

Mao, Junhua, Wei Xu, Yi Yang, Jiang Wang, Zhiheng Huang, and Alan Yuille. 2015. Deep captioning with multimodal recurrent neural networks (M-RNN). In *International conference on learning representations (iclr)*.

Miech, Antoine, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2020. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 9879–9889.

Miech, Antoine, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. HowTo100M: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the ieee/cvf international conference on computer vision*, 2630–2640.

Miller, George A. 1995. WordNet: a lexical database for english. *Communications of the ACM* 38(11):39–41.

Nevatia, Ram, Tao Zhao, and Somboon Hongeng. 2003. Hierarchical language-based representation of events in video streams. In *2003 conference on computer vision and pattern recognition workshop*, vol. 4, 39–39.

Nichol, Alexander Quinn, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. 2022. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In *International conference on machine learning*, 16784–16804. PMLR.

Oord, Aaron van den, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the association for computational linguistics (acl)*, 311–318. Association for Computational Linguistics.

Pei, Mingtao, Yunde Jia, and Song-Chun Zhu. 2011. Parsing video events with goal inference and intent prediction. In *2011 international conference on computer vision*, 487–494.

Pennington, Jeffrey, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)*, 1532–1543. Association for Computational Linguistics.

Peyre, Julia, Josef Sivic, Ivan Laptev, and Cordelia Schmid. 2017. Weakly-supervised learning of visual relations. In *Proceedings of the ieee international conference on computer vision (iccv)*, 5179–5188.

Pham, Hieu, Zihang Dai, Qizhe Xie, and Quoc V Le. 2021. Meta pseudo labels. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 11557–11568.

Plummer, Bryan A, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. 2015. Flickr30k entities: Collecting region-to-phrase

correspondences for richer image-to-sentence models. In *Proceedings of the ieee international conference on computer vision (iccv)*, 2641–2649. IEEE.

Qian, Yicheng, Weixin Luo, Dongze Lian, Xu Tang, Peilin Zhao, and Shenghua Gao. 2022. SVIP: Sequence verification for procedures in videos. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 19890–19902.

Radford, Alec, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning (icml)*.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Technical Report*.

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* 21(1):5485–5551.

Rahman, Shafin, Salman Khan, and Nick Barnes. 2020a. Improved visual-semantic alignment for zero-shot object detection. In *34th aaai conference on artificial intelligence (aaai)*.

Rahman, Shafin, Salman H Khan, and Fatih Porikli. 2020b. Zero-shot object detection: Joint recognition and localization of novel concepts. *International Journal of Computer Vision* 128(12):2979–2999.

Ramanathan, Vignesh, Rui Wang, and Dhruv Mahajan. 2021. Predet: Large-scale weakly supervised pre-training for detection. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 2865–2875.

Ramesh, Aditya, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with CLIP latents. *arXiv preprint arXiv:2204.06125*.

Redmon, Joseph, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 779–788.

Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. 2015a. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems (neurips)*, ed. C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, vol. 28. Curran Associates, Inc.

———. 2015b. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems (neurips)*, 91–99. Curran Associates, Inc.

Rennie, Steven J, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 7008–7024. IEEE.

Rohrbach, Anna, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. In *Proceedings of the 2018 conference on empirical methods in natural language processing (emnlp)*, 4035–4045. Association for Computational Linguistics.

Rombach, Robin, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 10684–10695.

Ryoo, M.S., and J.K. Aggarwal. 2006. Recognition of composite human activities through context-free grammar based representation. In *2006 ieee computer society conference on computer vision and pattern recognition (cvpr'06)*, vol. 2, 1709–1718.

Saharia, Chitwan, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*.

Sariyildiz, Mert Bulent, Julien Perez, and Diane Larlus. 2020. Learning visual representations with caption annotations. In *Eccv*.

Savva, Manolis, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. 2019. Habitat: A platform for embodied ai research. In *Proceedings of the ieee/cvf international conference on computer vision*, 9339–9347.

Schuster, Sebastian, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D. Manning. 2015. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the fourth workshop on vision and language*, 70–80. Lisbon, Portugal: Association for Computational Linguistics (ACL).

Selvaraju, Ramprasaath R, Stefan Lee, Yilin Shen, Hongxia Jin, Shalini Ghosh, Larry Heck, Dhruv Batra, and Devi Parikh. 2019. Taking a hint: Leveraging explanations to make vision and language models more grounded. In *Proceedings of the ieee international conference on computer vision (iccv)*, 2591–2600. IEEE.

Sener, Fadime, and Angela Yao. 2019. Zero-shot anticipation for instructional activities. In *Proceedings of the ieee/cvf international conference on computer vision*, 862–871.

Sener, Ozan, Amir R Zamir, Silvio Savarese, and Ashutosh Saxena. 2015. Unsupervised semantic parsing of video collections. In *Proceedings of the ieee international conference on computer vision*, 4480–4488.

Sharma, Piyush, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th annual meeting of the association for computational linguistics (acl)*, 2556–2565.

Shen, Yuhan, Lu Wang, and Ehsan Elhamifar. 2021. Learning to segment actions from visual and language instructions via differentiable weak sequence alignment. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 10156–10165.

Shetty, Rakshith, Marcus Rohrbach, Lisa Anne Hendricks, Mario Fritz, and Bernt Schiele. 2017. Speaking the same language: Matching machine to human captions by adversarial training. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 4135–4144. IEEE.

Shi, Jiaxin, Hanwang Zhang, and Juanzi Li. 2019. Explainable and explicit visual reasoning over scene graphs. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 8376–8384.

Shi, Jing, Yiwu Zhong, Ning Xu, Yin Li, and Chenliang Xu. 2021a. A simple baseline for weakly-supervised scene graph generation. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 16393–16402.

———. 2021b. A simple baseline for weakly-supervised scene graph generation. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*.

Simonyan, Karen, and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *Iclr*.

Singh, Bharat, Hengduo Li, Abhishek Sharma, and Larry S Davis. 2018. R-fcn-3000 at 30fps: Decoupling detection and classification. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 1081–1090.

Sohl-Dickstein, Jascha, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265. PMLR.

Sohn, Kihyuk, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. 2020. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*.

Song, Jie, Bjoern Andres, Michael J Black, Otmar Hilliges, and Siyu Tang. 2019. End-to-end learning for graph decomposition. In *Proceedings of the ieee international conference on computer vision (iccv)*, 10093–10102. IEEE.

Song, Kaitao, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems* 33:16857–16867.

Song, Yang, and Stefano Ermon. 2020. Improved techniques for training score-based generative models. *Advances in neural information processing systems* 33:12438–12448.

Su, Weijie, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2020. VL-BERT: Pre-training of generic visual-linguistic representations. In *International conference on learning representations (iclr)*.

Sun, Chen, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. 2019. Videobert: A joint model for video and language representation learning. In *Proceedings of the ieee/cvf international conference on computer vision*, 7464–7473.

Szegedy, Christian, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first aaai conference on artificial intelligence (aaai)*.

Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 1–9.

Tan, Hao, and Mohit Bansal. 2019. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*.

Tang, Kaihua, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. 2020a. Unbiased scene graph generation from biased training. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 3716–3725.

Tang, Kaihua, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019a. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 6619–6628.

Tang, Siyu, Bjoern Andres, Miykhaylo Andriluka, and Bernt Schiele. 2015. Subgraph decomposition for multi-target tracking. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 5033–5041. IEEE.

Tang, Yansong, Dajun Ding, Yongming Rao, Yu Zheng, Danyang Zhang, Lili Zhao, Jiwen Lu, and Jie Zhou. 2019b. Coin: A large-scale dataset for comprehensive instructional video analysis. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 1207–1216.

Tang, Yansong, Jiwen Lu, and Jie Zhou. 2020b. Comprehensive instructional video analysis: The COIN dataset and performance evaluation. *TPAMI*.

Teney, Damien, Lingqiao Liu, and Anton van Den Hengel. 2017. Graph-structured representations for visual question answering. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 1–9.

Tevet, Guy, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. 2023. Human motion diffusion model. In *International conference on learning representations*.

Tian, Zhi, Chunhua Shen, Hao Chen, and Tong He. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the ieee/cvf international conference on computer vision*, 9627–9636.

Touvron, Hugo, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017a. Attention is all you need. In *Advances in neural information processing systems* (*neurips*), vol. 30.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017b. Attention is all you need. *Advances in neural information processing systems* 30.

Vedantam, Ramakrishna, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (*cvpr*), 4566–4575. IEEE.

Vijayakumar, Ashwin K, Michael Cogswell, Ramprasaath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search for improved description of complex scenes. In *Aaai conference on artificial intelligence*.

Vinyals, Oriol, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the ieee conference on computer vision and pattern recognition* (*cvpr*), 3156–3164. IEEE.

Wald, Johanna, Helisa Dhamo, Nassir Navab, and Federico Tombari. 2020. Learning 3D semantic scene graphs from 3d indoor reconstructions. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (*cvpr*).

Wang, Alex Jinpeng, Yixiao Ge, Rui Yan, Yuying Ge, Xudong Lin, Guanyu Cai, Jianping Wu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. 2022. All in one: Exploring unified video-language pre-training. *arXiv preprint arXiv:2203.07303*.

Wang, Josiah, Pranava Swaroop Madhyastha, and Lucia Specia. 2018. Object counts! bringing explicit detections back into image captioning. In *Proceedings of the 2018 conference of the north American chapter of the association for computational linguistics* (*naacl*), 2180–2193. Association for Computational Linguistics.

Wang, Josiah, Katja Markert, and Mark Everingham. 2009. Learning models for object recognition from natural language descriptions. In *The british machine vision conference (bmvc)*, vol. 1, 2.

Wang, Limin, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, 20–36. Springer.

Wang, Liwei, Alexander Schwing, and Svetlana Lazebnik. 2017. Diverse and accurate image description using a variational auto-encoder with an additive gaussian encoding space. In *Advances in neural information processing systems (neurips)*, 5756–5766. Curran Associates, Inc.

Wang, Wenbin, Ruiping Wang, Shiguang Shan, and Xilin Chen. 2020. Sketching image gist: Human-mimetic hierarchical scene graph generation. In *European conference on computer vision (eccv)*. Springer.

Wani, Saim, Shivansh Patel, Unnat Jain, Angel X. Chang, and Manolis Savva. 2020. Multi-on: Benchmarking semantic map memory using multi-object navigation. In *Neural information processing systems (neurips)*.

Weihs, Luca, Matt Deitke, Aniruddha Kembhavi, and Roozbeh Mottaghi. 2021. Visual room rearrangement. In *Ieee/cvf conference on computer vision and pattern recognition (cvpr)*.

Wu, Yuxin, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. 2019. Detectron2. `https://github.com/facebookresearch/detectron2`.

Xie, Qizhe, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. 2020. Self-training with noisy student improves imagenet classification. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 10687–10698.

Xie, Saining, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. 2018. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the european conference on computer vision (eccv)*, 305–321.

Xu, Danfei, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 5410–5419.

Xu, Frank F., Lei Ji, Botian Shi, Junyi Du, Graham Neubig, Yonatan Bisk, and Nan Duan. 2020. A benchmark for structured procedural knowledge extraction from cooking videos. In *Proceedings of the first international workshop on natural language processing beyond text*, 30–40. Online: Association for Computational Linguistics.

Xu, Hu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. 2021a. VideoCLIP: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*.

Xu, Kelvin, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning (icml)*, 2048–2057.

Xu, Keyulu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. 2019. How powerful are graph neural networks? In *International conference on learning representations (iclr)*.

Xu, Mengde, Zheng Zhang, Han Hu, Jianfeng Wang, Lijuan Wang, Fangyun Wei, Xiang Bai, and Zicheng Liu. 2021b. End-to-end semi-supervised object detection with soft teacher. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 3060–3069.

Yalniz, I Zeki, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. 2019. Billion-scale semi-supervised learning for image classification. *arXiv preprint arXiv:1905.00546*.

Yan, An, Yu Wang, Yiwu Zhong, Chengyu Dong, Zexue He, Yujie Lu, William Yang, Jingbo Shang, and Julian McAuley. 2023. Learning concise and descriptive attributes for visual recognition. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*.

Yang, Jianwei, Yonatan Bisk, and Jianfeng Gao. 2021. Taco: Token-aware cascade contrastive learning for video-text alignment. In *Proceedings of the ieee/cvf international conference on computer vision*, 11562–11572.

Yang, Jianwei, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018a. Graph R-CNN for scene graph generation. In *Proceedings of the european conference on computer vision (eccv)*, 670–685.

———. 2018b. Visual curiosity: Learning to ask questions to learn visual recognition. In *Conference on robot learning (corl)*.

Yang, Ling, Zhilong Zhang, Yang Song, Shenda Hong, Runsheng Xu, Yue Zhao, Yingxia Shao, Wentao Zhang, Bin Cui, and Ming-Hsuan Yang. 2022. Diffusion models: A comprehensive survey of methods and applications. *arXiv preprint arXiv:2209.00796*.

Yang, Linjie, Kevin Tang, Jianchao Yang, and Li-Jia Li. 2017. Dense captioning with joint inference and visual context. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 2193–2202. IEEE.

Yang, Xu, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019. Auto-encoding scene graphs for image captioning. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 10685–10694.

Yao, Ting, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *Eccv*, 684–699.

Ye, Keren, and Adriana Kovashka. 2021. Linguistic structures as weak supervision for visual scene graph generation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 8289–8299.

Ye, Keren, Mingda Zhang, Adriana Kovashka, Wei Li, Danfeng Qin, and Jesse Berent. 2019. Cap2Det: Learning to amplify weak caption supervision for object detection. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 9686–9695.

Yin, Xuwang, and Vicente Ordonez. 2017. Obj2Text: Generating visually descriptive language from object layouts. In *Proceedings of the 2017 conference on empirical methods in natural language processing (emnlp)*, 177–187. Association for Computational Linguistics.

You, Quanzeng, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. 2016. Image captioning with semantic attention. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 4651–4659. IEEE.

Yu, Fei, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2021. Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. In *Proceedings of the aaai conference on artificial intelligence*, vol. 35, 3208–3216.

Yu, Ruichi, Ang Li, Vlad I Morariu, and Larry S Davis. 2017. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the ieee international conference on computer vision (iccv)*, 1974–1982.

Yuan, Lu, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. 2021. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*.

Zareian, Alireza, Svebor Karaman, and Shih-Fu Chang. 2020. Weakly supervised visual semantic parsing. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 3736–3745.

Zareian, Alireza, Kevin Dela Rosa, Derek Hao Hu, and Shih-Fu Chang. 2021. Open-vocabulary object detection using captions. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 14393–14402.

Zellers, Rowan, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. 2021. Merlot: Multimodal neural script knowledge models. *Advances in Neural Information Processing Systems* 34.

Zellers, Rowan, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 5831–5840.

Zhang, Hanwang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. 2017a. Visual translation embedding network for visual relation detection. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 5532–5540.

Zhang, Hanwang, Zawlin Kyaw, Jinyang Yu, and Shih-Fu Chang. 2017b. PPR-FCN: Weakly supervised visual relation detection via parallel pairwise R-FCN. In *Proceedings of the ieee international conference on computer vision (iccv)*, 4233–4241.

Zhang, Ji, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. 2019. Graphical contrastive losses for scene graph parsing. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 11535–11543.

Zhang, Pengchuan, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. 2021. Vinvl: Revisiting visual representations in vision-language models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*, 5579–5588.

Zhao, Henghui, Isma Hadji, Nikita Dvornik, Konstantinos G. Derpanis, Richard P. Wildes, and Allan D. Jepson. 2022. P3IV: Probabilistic procedure planning from instructional videos with weak supervision. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* 2928–2938.

Zhong, Yiwu, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. 2021. Learning to generate scene graph from natural language supervision. In *Proceedings of the ieee/cvf international conference on computer vision (iccv)*, 1823–1834.

Zhong, Yiwu, Liwei Wang, Jianshu Chen, Dong Yu, and Yin Li. 2020. Comprehensive image captioning via scene graph decomposition. In *Proceedings of the european conference on computer vision (eccv)*, 211–229. Springer.

Zhong, Yiwu, Jianwei Yang, Pengchuan Zhang, Chunyuan Li, Noel Codella, Liunian Harold Li, Luowei Zhou, Xiyang Dai, Lu Yuan, Yin Li, et al. 2022. Regionclip: Region-based language-image pretraining. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 16793–16803.

Zhong, Yiwu, Licheng Yu, Yang Bai, Shangwen Li, Xueting Yan, and Yin Li. 2023. Learning procedure-aware video representation from instructional videos and their narrations. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 14825–14835.

Zhou, Bolei, Alex Andonian, Aude Oliva, and Antonio Torralba. 2018a. Temporal relational reasoning in videos. In *Proceedings of the european conference on computer vision (eccv)*, 803–818.

Zhou, Luowei, Yannis Kalantidis, Xinlei Chen, Jason J Corso, and Marcus Rohrbach. 2019. Grounded video description. In *Proceedings of the ieee conference on computer vision and pattern recognition (cvpr)*, 6578–6587. IEEE.

Zhou, Luowei, Hamid Palangi, Lei Zhang, Houdong Hu, Jason Corso, and Jianfeng Gao. 2020. Unified vision-language pre-training for image captioning and vqa. In *Proceedings of the aaai conference on artificial intelligence*, vol. 34, 13041–13049.

Zhou, Luowei, Chenliang Xu, and Jason J Corso. 2018b. Towards automatic learning of procedures from web instructional videos. In *Thirty-second aaai conference on artificial intelligence*.

Zhou, Xingyi, Vladlen Koltun, and Philipp Krähenbühl. 2021. Probabilistic two-stage detection. *arXiv preprint arXiv:2103.07461*.

Zhu, Deyao, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

Zhu, Linchao, and Yi Yang. 2020. ActBERT: Learning global-local video-text representations. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 8746–8755.

Zhu, Pengkai, Hanxiao Wang, and Venkatesh Saligrama. 2020. Don't even look once: Synthesizing features for zero-shot detection. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition (cvpr)*.

Zhukov, Dimitri, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. 2019. Cross-task weakly supervised learning from instructional videos. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 3537–3545.

Zoph, Barret, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. 2020. Rethinking pre-training and self-training. *Advances in Neural Information Processing Systems* 33.